

DESIGNING AN OBJECT-ORIENTED GEOGRAPHIC INFORMATION SYSTEM
APPLICATION FOR INTEGRATION OF BUILDING CONSTRUCTION PROJECT
INFORMATION

By

WEI SUN

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2001

Copyright 2001

by

Wei Sun

ACKNOWLEDGMENTS

My sincere gratitude and appreciation go to my supervisory committee for their guidance throughout this research and preparation of this dissertation. Many heartfelt thanks belong to Dr. John Alexander for his tremendous contribution and consistent support. His excellent guidance is extremely beneficial, as is his extensive technical expertise. His kind mentorship and sage advice are always treasured. His wisdom and excellence in matters both academic and professional are always a true source of inspiration. My gratitude also goes to Dr. Jimmie Hinze for his knowledgeable advice, valuable guidance and constant motivation. Sincere appreciation goes to Dr. Paul Zwick for his insightful ideas and wise counsel. Special acknowledgement goes to Dr. Pierce Jones for his scholarly advice and insights. A very special thanks is owed to Dr. Mark Schmalz. His detailed review of dissertation is especially helpful and very much appreciated. His guidance with the research and with curriculum is also very much appreciated.

I am also very much obliged to Dr. Jo Hassel for her unfailing support and encouragement throughout my time at University of Florida. Gratitude is also extended to Ms. Susie Studtill of Ph.D. program in College of Design, Construction and Planning for her cheerful support and motivation.

TABLE OF CONTENTS

	<u>Page</u>
ACKNOWLEDGMENTS.....	iii
ABSTRACT	vi
CHAPTERS	
1 INTRODUCTION.....	1
Background	1
Literature Review	6
Research Objectives	19
2 METHODS AND MATERIALS	32
Project Modeling	33
Georelational Modeling.....	38
Object-centered Modeling.....	43
3 RESULTS.....	47
Project Information Modeling Development	47
System Implementation of Construction GIS	73
System Case Testing	95
4 DISCUSSION AND SUMMARY	119
Research Findings	119
Research Assessment	128
Future Development.....	131
APPENDIX A PROJECT OBJECT LIBRARY SPECIFICATION.....	134
APPENDIX B CONSTRUCTION GIS SYSTEM DESCRIPTION	155
LIST OF REFERENCES	179

BIOGRAPHICAL SKETCH	187
---------------------------	-----

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

DESIGNING AN OBJECT-ORIENTED GEOGRAPHIC INFORMATION SYSTEM
APPLICATION FOR INTEGRATION OF BUILDING CONSTRUCTION PROJECT
INFORMATION

By

WEI SUN

May 2001

Chairman: Dr. John F. Alexander

Major Department: College of Design, Construction and Planning

The complexity and diversity of the vast amount of information in construction management requires a coordinated and consistent system that enhances full integration of construction information. This research effort is objected towards the development of a GIS (Geographic Information System) based integrated information system for construction project information integration.

This research develops a framework for an integrated project information management system, based on GIS technology and a GIS based tools. The system can potentially enables project participants to access, navigate and manipulate information typically used in construction projects to support the decision making process during various project phases in an integrated environment. A research approach has been adopted for modeling and communicating information about a construction project using

three components: a geo-relational modeling kernel, a multi-layered structure of underlying project modeling base, and the object-based procedural interface.

The GIS-based information integration system has been developed in three stages. System development starts with the construction of an object-centered, GIS-based geo-relational modeling framework based on a systematic approach. It is considered as the hub of the proposed methodology through which information integration is achieved. The next stage involves the system implementation that aims to support the information requirements of the information integration process using appropriate system architecture. The specification of the system approach along with the mechanism that supports integration is incorporated into the GIS-system. The third stage is to put the modeling framework and system approach into a prototype system. A prototype system, named as Construction GIS is developed and put into test. The feedback from the testing is used for refining the proposed methodology.

The research demonstrated the adequate functions of GIS-based integrated information systems for getting relevant product data technology in place to effectively support construction project information integration. Methods used in this research also suggested improving the efficiency of the linkage between GIS and models include implementing various database management strategies and interfaces within a GIS environment.

CHAPTER 1 INTRODUCTION

The complexity and diversity of the vast amount of information in construction management requires a coordinated and consistent system that enhances full integration of construction information. The focus of this research is to demonstrate the potential of a GIS (Geographic Information System) in construction project information integration schemes. The overriding perspective of the research is to apply object oriented programming with GIS technology to build an integrated framework and system for construction project information integration.

Background

The construction process includes many participants, who create, change, analyze, evaluate and comment on a large volume of shared and proprietary information. Current construction management processes rely heavily on information produced by others in separate phases or views of a project, and are dependent on the quality, efficiency, and timeliness of various required information inputs. Unfortunately, most professionals involved in building construction projects consider these myriad systems as islands; i.e. there are 'information barriers' between professionals involved in construction projects (Hannus 1992).

In the case of design and engineering data used by construction professionals, a wealth of electronic information in technical databases has been produced. Much of these data can be extremely valuable to the construction team. However, the content, structure, and scope of the data produced are often incompatible with the way in which construction professionals view a project (Stumpf, Liu, Chin & Ahn 1994). In some cases, construction may need more abstraction, in others much less abstraction. Usually, the partial data from multiple engineering sources must be grouped together to support a single construction task. Likewise, different pieces of data from a single engineering source may need to find their way to different users in the construction team.

Inside the construction management domain, the same problems are encountered. Cost, schedule and control are interrelated in construction management. But cost estimation, scheduling and control systems develop data independently. Different procedures among various professionals cause a lack of information sharing among project management teams. Cost estimation is based upon a resource-oriented data structure in which cost is segregated by resource type such as concrete, masonry or metal. Scheduling data is based on a system-oriented data structure such as foundation, substructure and superstructure. The two different data structures lead to a fundamental information barrier between cost estimation and scheduling. Because of the disconnection, each professional such as the estimator or scheduler has certain difficulties in the acquisition of data. Much added work is required in transferring commonly needed information between these functional areas. Accuracy and completeness are compromised in the process, and the issues of data ownership and responsibility can be difficult to resolve. Serious problems arise when the available information is to be

implemented, due to difficulties in retrieving information required and in examining related items. It takes commitment and resources to accommodate such changes so as to maintain compatibility among all construction information.

Furthermore, the dynamic nature of construction information adds another concern to construction integration. In construction, massive amounts of information are dramatically modified and increased as the project develops. Due to the complexity and dynamic nature of construction projects, changes during the construction process are inevitable. As construction information is interrelated, change on a single item of project information may affect many related components. On any construction project, any change must be collected and processed by a variety of parties in order to control the process and protect their particular interests. Industry practice in communicating and resolving information change in construction projects is far from satisfactory (Pena-Mora & Chen 1997). Data are not well organized, flexible, or easily accessible. Additionally, change information is often not provided in a useful format or timely basis. Therefore mistakes and inconsistencies are inevitable. This can cause a lot of controversies, despite the help of computer project management systems.

Besides information processing and retrieving, another challenge facing construction information management is information analysis for early detection of problems, identification of their source, and selection of appropriate corrective actions. This challenge is increased by the vast volumes of data that project management staff must synthesize in order to maintain up-to-date views of a project. This issue requires in-depth analysis of existing conditions and anticipation of the major trend that are occurring within the construction project and the possible effect as to the utilization of the

end product. In developing a strategy to solve problems on construction sites, the problem solver has to draw upon and process a wide spectrum of complex, inter-related information from the external world and link it with his or her own judgement and knowledge about technology. To support these functions, information must be available to convert miscellaneous transactions into meaningful interpretations of patterns and trends. Current decision control process has shortcoming in terms of information communications and joint decision making with the project team members. The following list provides several shortcomings:

- Construction information cannot be communicated and exchanged among different organizations and participants.
- Project information cannot be presented or summarized at multiple levels of abstraction.
- Complicated interdependencies of construction information cannot be addressed.
- The ripple impacts of information change cannot be managed.
- Information might not be available to convert miscellaneous transactions into meaningful interpretations of patterns and trends to support the decision-making.

Collection and management of various types of information efficiently and on demand, is seen as the key to successful management of construction projects. Total information integration will improve communication among computer systems that will, in turn, lead to better information sharing among projects participants. The visions that follow from the integration of diverse information are widely recognized (Aouad et al. 1994):

- A broader base of information can be drawn upon that allows one to address issues which were previously beyond their individual data resources.
- Project teams can cooperate with one another within the context of integrated information, and thereby make more effective management decisions.
- Systematic management, exchange and sharing of information make the information process more efficient because of less risks for misinterpretations, less cost of data acquisition and less need for redundant repeated manual data input.
- A broader range of operations can be performed on integrated information than on disparate sets of data, thus providing a more solid foundation for decision making.
- Through the integration of data which were previously the domain of individual disciplinary specialists, an interdisciplinary perspective to problem solving is encouraged, and more solid decision making can result.
- Users benefit from the perception that they have access to a seamless information environment, uncomplicated by the need to consider differences in data sources, information types, storage devices, computer applications, etc.

It may be that the most valuable benefit of information integration will come from improvements in the availability and use of information, especially considering the impact on decision making in the early stages of building projects. Information integration allows for increased efficiency and accuracy while providing improved decision-making capability. Not only will information integration reduce errors and inefficiencies resulting from inaccurate, untimely, or missing information, but it will also help foster better coordination and cooperation of construction teams. These benefits

could be related directly to reducing construction project cost, increasing constructability and improving project job performance. In the long term, it means a great deal of economical benefit along with improved management benefit.

Literature Review

This literature review summarizes the work, both past and present, that has been done in the area of information integration within the construction industry. Many studies have identified the construction industry as a multi-actor cooperative process which could greatly benefit from IT, and possibly as the main key-enabler of integration for a presently fragmented industrial process (Fischer 1989; Hannus 1992; Aouad et al. 1994; Laitinen 1995; Underwood and Alshaw, 1996; Amor et al. 1997; Aouad 1997; Wix 1998; Bjork, 1999; Lottaz et al. 2000). The development and deployment of new construction industry software applications, improvements in network technology, the development of new modeling methodologies, and the definition of standards for information exchange all create new opportunities for information integration within the construction industry.

Construction Information Integration

There exists no generally accepted definition of the word 'integration' in the context of construction information management. Researchers at the Center of Integration Facility Engineering at Stanford have defined integration as the continuous interdisciplinary sharing of data, knowledge, and goals among participants. To be more specific, information integration means to reconcile a variety of information elements

representing disciplinary and building components through time (Fischer 1989). Generally, information integration is based on the concept of vertically as well as horizontally integrating many data, functions and knowledge based among participants of a project (Laitinen 1995), as follows:

- *horizontally* in the sense of managing the creation, revision and exchange of information carried out simultaneously by several partners; and
- *vertically* in the sense of transferring information from one process phase to another.

Aouad et al. (1994) stated that integration concept should consider to be based on five main strategies. These integration strategies must include:

- Project team, including both inter-functional and intra- functional disciplines;
- Project evolution processes, including every phase of project development;
- Project activities, including every professional activity such as cost estimating, scheduling and project control;
- Project data, including both textual, graphic and multimedia data; and
- Project tools, including all construction computing applications.

It can be readily seen that there are many aspects of information integration in construction industry, including both technical and non-technical aspects (Eastman 1999). Change can only occur when both the organizational structure and the information tools are improved. Since the non-technical organizational changes are often significant

and difficult to overcome, it make sense to approach information integration by first solving technical problems. Building a sophisticated system for information integration in a construction project process may make it possible to accommodate the various needs of different building professionals. Numerous reports by research establishments (STEP, 1999; IFC 1999) on the future usage of IT in the construction industry, concluded that there is a great need for the development and implementation of integrated applications to allow for future sharing of project information. Previous researchers have proposed hypotheses and strategies on how the emergence of information technology can successfully achieve the construction information integration (Froese 1999).

To address the technical issue of information integration, it is required that integrated systems be able to take the information produced by the multiple sources and easily reconfigure and then deliver that information to construction management team in a form consistent with construction information integration requirements. The integrated information system is frequently described as the synthesis of construction information in a computer system, which depends on its effectiveness on information linkage (i.e. of textual and geometric data) within a coherent data model (Russell & Froese 1995). This involves bringing together diverse information from a variety of sources (information interchange), requires the effective matching of supposedly similar entities in these sources, and demands information consistency across the source data sets. Any integrated system should provide a means by which the component objects provided by organizations and application vendors can interoperate in a seamless way. This is what the integrated information system is all about: providing the 'glue' between the different information pieces that support the work in different process stages (Russell & Froese

1995). The common approach of integrated systems is an increase in the use of current and emerging information technologies. Advances in this area have been considerably based on the recent IT progress in object-oriented databases, distributed databases, networks, transaction mechanisms, standard exchange languages and integration services for heterogeneous systems.

Integration has been at the forefront of research agendas in the past few years. The development of integrated construction information systems has been the goal of many research efforts. Of these efforts, some are based on extensively developing construction information models while others have been attempting to realize an integrated system using product models or project models. The following is a listing of prominent ongoing related research projects:

ISO STEP

The increasing digitalization of the information related to industrial projects has led ISO (International Organization for Standard) organization, through its sub-committee 4 of Technical Committee 184, to undertake the development of international standards. Work within ISO's Standard for Exchange of Product Model Data (ISO STEP) is developing international standards dealing with the use of digital product and manufacturing management data. The STEP project, referred to as ISO 10303, is an international standard for the computer-interpretable representation and exchange of product data, with the objective of providing a neutral mechanism to describe the product data through the life cycle of the product independent of any particular system (ISO

1994a). The completeness of this mechanism makes it suitable not only for neutral file exchange, but also as a basis for implementing and sharing databases and archiving. Work within ISO STEP includes the development of standards to specify generic resources and methods for representing libraries of standardized product descriptions, the development of resource models (including generic resources, application resources and application protocols), the development of methods that will facilitate product information description and exchange. STEP is a collection of standards called Application Protocols. The Application protocol development starts with the construction of the AAM (Application Activity Model) using the IDEF0 methodology. The AAM defines the application's scope and activity decomposition. The next stage involves the ARM (Application Reference Model), which supports the information requirements of the AAM using appropriate languages such as NIAM, EXPRESS-G and EXPRESS. Integration is then achieved through the AIM (Application Integrated Model) which involves using and specializing the integrated information resources (provided by STEP) to facilitate communication between different disciplines and sectors of industry. The overall APs covering the AEC domains are developed and integrated within a common frame. Of particular interest here is Part 106 Building Construction Core Model (BCCM) (ISO 1996). The STEP community is developing tools like EXPRESS-X and the Standard Data Access Interface (SDAI) in Java to access, query and retrieve STEP instance data from distributed locations. Research projects using STEP or STEP-related technologies that are of relevance to building and construction include CONDOR (CONstruction project Documentation pRoduction and management), EIEM (Engineering Information Management Executive), ELSEWISE (European Large Scale

Engineering Wide Integration Support Effort), ToCEe (Towards a Concurrent Engineering Environment), VEGA (Virtual Enterprise using Groupware tools and structured Architecture), etc. Informally, STEP is a methodology and set of technologies for industry professionals to agree upon object meaning and description in an unambiguous, neutral and machine-intelligible format. It provides the methodology for construction information integration but does not target the implementing the computerized integrated information system for practical use in building construction project management.

IAI and IFCs

The Industry Alliance for Interoperability (IAI) is a global, industry-based consortium for the AEC/FM industry (IAI 2000). Their mission is to enable interoperability among industry processes of all different professional domains in AEC/FM projects by allowing the computer applications used by all project participants to share and exchange project information (IAI 1998). The IAI's scope is the entire lifecycle of building projects including strategic planning, design and engineering, construction, and building operation. The IAI's goals are to define, publish and promote a specification--called the Industry Foundation Classes (IFCs)--for sharing data throughout the project lifecycle, globally, across disciplines and technical applications (IAI 2000). IFC efforts are closely related with the STEP effort and methodologies. In fact, the BCCM and the IFC core classes are currently being co-developed to create a unified core model shared between the two efforts. The IFCs are object classes that represent information about a project and are used to construct project model. The models are

shared by the computer applications used by project participants to facilitate communication through a commonly understood language defined in the models (Yu and Grobler, 1998). IFC support the use of information standardization, as a means to provide an opportunity to address the sources of conflict, separation of functions, as well as the "over-the-wall" sequence of activities which lead to the arbitrariness in the construction industry. Much of the IFCs' focus has been on representing the facilities that are being designed and constructed, but the IFCs' scope also includes project management information such as costs, schedules, work tasks, resources, etc. The IFCs are developed through IAI projects targeted at specific IFC releases. Several IFC projects undertaken by different regional domain committees focus on project management related aspects of the IFCs. The Project Management (PM) Domain Group of the IAI North American Chapter (IAI NA PM) has been developing portions of the IFCs to support estimating (Project ES-1 for IFC Release 2.0) (Cole et al. 1998) and scheduling (PM-1 for IFC Release 3.0) (Grobler et al. 1998, Froese and Yu 1999) and their integration for project management (Froese etc. 1999; Yu, Froese and Grobler 2000). The IFCs are used to assemble a project model in a neutral computer language that describes building project objects and represents information requirements common throughout all industry processes. Its focus mainly on the building the core model which is intended to be interpreted by application of integrated information system. Until now very few systems have been implemented with some application of IFC PM objects (Laitinen 1999).

TOPS

Total-Project Systems (TOPS) research program within the University of British Columbia's Construction Management and Engineering group, is an approach to integrated, computer-based tools to support construction management (Froese, Rankin and Yu, 1997). TOPS attempts to link a number of application programs by linking systems and providing data transfer interfaces to reach the objective of integration (Froese and Rankin, 1998). Since the projects' conception, TOPS have been succinctly described by three primary characteristics: comprehensive, integration and flexibility (Froese, T., J. Rankin and K. Yu, 1997). The vision of integration is that all TOPS applications both contribute to and draw from a common pool of project information. Through the TOPS interface, a group of construction management application tools are available for use. One of the major components of the TOPS approach is the common data model of construction projects upon which the system is based. The common data model is significant since it provides the primary mechanism for structuring data within TOPS applications, for exchanging information among TOPS components, and for interacting with other computer applications through international data standards. Furthermore, structuring construction project data according to a fairly high-level semantic model imparts greater computer-interpretable meaning to the data, which contributes to the functionality of the system. A second major element of TOPS is the development of the various application modules that make up the integrated system. TOPS is intended to support both the traditional and emerging construction management applications. The architecture allows for a "plug and out" feature. The user-interface employed is based on a commonly accepted 2-dimensional "tree and detail" scheme within MDI (multi-document interface) approach. (Rankin, Forese and Waugh, 1999).

The implementation of 3D and more elaborate interfaces are being pursued in a concurrent effort at the University of New Brunswick (Rackley et al. 1998). TOPS and related projects present a unique information integration approach in a way it approaches information integration through an intelligent interface among heterogeneous applications. In fact, TOPS builds communication layers among application programs and common database but it does not build the integrated system as a standalone program.

4D-CAD

4D-CAD research at the Center of Integrated Facility Engineering (CIFE) at Stanford University has shown the benefits and opportunities of achieving integrated system through geometry (CIFE 2000). 4D CAD (3D geometry plus time) would allow users to explore and interlink a wide variety of construction project information, together with process design and analysis tools, to engineer efficient construction process plans. These new process design tools could be incorporated within scheduling tools, estimating tools, CAD tools, or they could form a whole new application class. 4D CAD links a 3D graphic model and a construction schedule through the 4D CAD interface. The linking between CAD and schedule graphically incorporates facility design with the information traditionally represented in the construction schedule. By doing so, the temporal and physical aspects of the project are integrated (Fisher 2000). Implementation of these concepts in 4D CAD tool involved AutoCAD representing the user interface and graphic model and Design Power's Design ++ (D++) system representing the symbolic product and process model. D++ is a knowledge-based engineering design system that provides

object-oriented representation and model-based reasoning with tight links to AutoCAD (McKinney, Kim, Fischer & Howard, 1996). The nearest development of 4D CAD modeling already goes beyond the original perspective. The implication is that 4D CAD systems could shift emphasis towards accommodating wide range of long and short term performance dimensions, that is nD CAD (Barrett, 2000). The benefits of n-dimensional modeling have been demonstrated through experimentation, and include the identification of potential conflicts between building elements and work spaces (Arkinici and Fischer, 1998 & 2000), document changes in job site conditions (Coble et al. 2000), safety hazards created due to proximity of construction activities, trade sequencing and production planning (Riley 2000), and visualization of construction plans by crews (Fischer 1998). One significant effect of 4D CAD is that it has established the role of using spatial-based building models for design and communication in building projects and achieved information integration through geometry. It provides the basic approach of building integrated system based on geometry but the tool for their use is CAD system not the GIS system. CAD system has its power of dealing with spatial data. But as we concerned, GIS has an even better capability of dealing with spatial data and non-spatial data as well.

My research project initially addressed and emphasized the technical platform for integrated information system, as opposed to previous researches as STEP and IFC which aimed to developing standard product models. Building product models, as well as industry data model standards for product models such as the Industry Foundation Classes (IFC's), have matured significantly and product models are successfully used to support some applications. Integrated information systems have not entered mainstream

use and many unresolved technical difficulties remain (Aouad 1997). The goals of my study are not to add to information technology as such; rather it targets efficient use of advanced GIS (Geographic Information System) technology in the building industry showing its potential to achieve the core concept of information integration.

However, my study utilizes ideas from previous researches both on standard models and systems. A complex product model is the most basic gradient for any fully integrated systems. For my research, STEP and IFC are important source of modeling and implementation methodologies. This research adopts modeling concepts from STEP and the IAI IFC's where they already address the appropriate bodies of information, and attempts to contribute to these efforts where new model development is required. My research also adopts the necessary methodologies for building integrated information system from previous research such as 4D CAD and TOPS. But we would emphasize that my research, built upon construction information integration domain perspectives, will meet its own specific aims to support information integration with a specific technical medium of GIS in mind.

Geographic Information System (GIS)

GIS is among the most widely embraced software technologies of the past decade. For many people, GIS is "mapping software". A GIS creates maps from data pulled from database. Formally, GIS is a powerful database technology for the management of data having a spatial character. Digital map products can then be created showing selected information symbolized effectively to highlight specific characteristics. In a stricter sense, a GIS is a computer system capable of assembling, storing, manipulating, and

displaying geographically referenced information, i.e. data identified according to their locations. dimension. One of the main benefits of GIS is improved management of information resources. A GIS can use information from many different sources, in many different forms and can link data sets together by common locational data, such as addresses. A GIS makes it possible to link information that is difficult to associate through any other means. Thus, a GIS can use combinations of information to build and analyze integrated information. A GIS can also convert existing digital information into a form that meets user's analysis need. Visualization of information analysis result is important benefit of GIS as it presents facts in a compelling way. The information can be presented succinctly in the form of a map and accompanying report, allowing clear information understanding. The old adage "better information leads to better decisions" is true for GIS. A GIS is not just an automated decision making system but a tool to query, analyze, and map data in support of the decision making process (ESRI 2000).

GIS applications are becoming common in diverse areas such as facilities location and planning, site selection and preparation, land management, road planning, management and design, environmental monitoring and analysis, residential and commercial site surveying, public works surveys and engineering, municipal and utility surveys, infrastructure evaluation, soils modeling, and etc. The rapid evolution of GIS technology over the past decade has motivated major GIS development efforts ranging in scale from very local to global. The rationale behind the development of these GIS applications is that accurate and reliable spatial representation of data will support better, or at least more efficient, decision-making.

For construction industry, however, the technology of GIS does not have a wide range of applications. The GIS application in construction industry is limited. While reliable databases are certainly an important component of an integrated information system, the integration of proven project modeling and information analysis methodologies is also essential (Wright, 2000). Unfortunately, the integration of GIS and conventional construction project modeling methods has not evolved to the point where information analysis is widely conducted using spatially oriented decision-support systems. Until we are able to achieve a high level of models-GIS integration, the benefits of GIS will not be reaped in full.

For most of previous research projects on integrated information system, CAD applications are used and so are database applications. Since CAD is designed for facility design, it provides a good package of graphic capability but lacks data management functions. Also since the information in the database does not include graphic representations, it is hard to browse objects. To resolve the practical limitation of both CAD and database systems, many research projects on integrated information system have developed integrated CAD-database programs that link CAD graphical elements to database records. These types of applications indeed integrate the graphics with the data to some degree, but they have to build special utility to link graphical and non-graphical data. GIS applications provide similar graphical interfaces just as CAD, but in addition extend the capability of associating non-graphic data with graphics, and incorporating the database applications too. This approach is superior to either a CAD or database application. This study introduced a natural progression towards increasing semantic information (i.e., CAD entities modeled as specific building components rather than

generic shapes) and increasing non-geometric information in the CAD models – that is, towards the use of GIS-based information integration.

Research Objective

This research is to provide a window into the future of how GIS can be applied in the construction information integration. It is anticipated incorporating ideas from mainstream project modeling efforts and integrated system approaches. But it is implemented in a way that allows the dynamic development of the modeling schema in order to support ongoing work in the development of information model standards with GIS capability. This domain perspective, together with the technical platform perspectives, aligns with the research objective of a GIS-based integrated information system for construction project management. This research is a GIS-based integration strategy to explore a possible solution with a GIS-based integrated information system, and it meets the challenge of construction project information integration in a unique way.

Research Questions and Challenges

A major research question at the outset is whether a GIS approach to information integration in the construction industry could be useful. In order to demonstrate that GIS is capable of the desired solution, the key research tasks remains to propose and implement a pragmatic approach leading to construction project information integration based on GIS technology.

To answer these research questions, certain research challenges need to be resolved. An integrated system that responds to the realities of the working environment in construction management must be developed. Several research challenges have been raised by previous researches (Russell and Froese, 1997). Among these are information environment, data representation, function set and system configuration challenge. These are discussed as follows.

The first challenge of my study is to identify supporting information that should be included within an integrated system. Because of the integrated system's characteristic of comprehensiveness, the scope of the underlying information must be complete and expansive. Generally speaking, the intended scope of the information set should cover all of the viewpoints of professionals included in construction management team. Approaches to construction information integration generally call for shared databases of information across the participant boundaries of the industry. One difficulty with this breadth requirement is the complexity of construction project information. Similarly, the supporting data structures share this characteristic. Construction project data is not only profuse but also it is collected in complex formats, including textual, graphical and multimedia formats. Thus, information supported by the integrated system should include CAD file, cost and schedule data, field survey data, photogrammetry and others. The ideal integrated system should provide an information framework that is generic, flexible and organized with each piece of data residing in only one location. The integrated information system is intended to implement a construction database structure using a generic data model. A generic database structure means that it should be possible to store and manage different kinds of data with the same database structure. The data structure of

the integrated construction project database should be possible to store and handle any kind of component data. A multimedia capability is required for accessing and interacting with complex facilities data. These would include scanned in on-the-site photographs, drawings, finishing schedules as well as parameters models that describe project scale and physical system characteristics. The system must be flexible enough to be tailored to any construction project data and provide sufficient data structure to support different project information.

A second and fundamental challenge to my study deals with definition of the various representations supported by the integrated system. In order to provide an organization with an effective information system, a holistic data representation approach toward data and information is required. In the construction industry, the need for integration between graphical and non-graphical data is highlighted. Existing software systems that support the project information management generally provide specific data representation. Most of them are also based mainly on geometry or they are analysis oriented, without adequate representation for problem definition, requirements processing and conceptual design. In addition, most systems or programs do not have the capability to capture the integration of different project information. Challenges arising from these described priorities deal with both the definition and the support of project data representations, mappings to connect representations, and the ability to work with information in large chunks. The ability to associate one group of representations (e.g., photos, building parameters) with other representations (e.g., activity and pay item) of the project is essential. Thus, it is more powerful to use a GIS-based integrated information

system to provide formalism for representing loosely structured project information of numerous types and varieties.

A third basic challenge to my study is to identify the function set of the information system. If a system is limited to areas of project management that are currently well-supported by computer tools, the scope of the integrated information system would be significantly reduced. For such a system to succeed in practice, it must integrate with CAD, perform specification editing, scheduling and cost estimating packages and/or provide the same functionality. The preferred scenario is an integrated, graphical, easy-to-use environment that provides intelligent control over the utilization of information and delivers different views of the integrated data in a variety of formats. Importantly, the information in the system must be organized such that it will be useful when retrieved. Access to information in the system must be managed and carefully regulated. There must be continued support and maintenance of the information within the construction project over time. A practical system must also support the administrative procedures for information management. Enhanced usability of the system should also arise from the ability to incorporate decision support capabilities in systems so that information can be manipulated to meet different professional needs.

A fourth challenge to my study is to define a system configuration and architecture for an integrated system. Currently, the emphasis of the majority of present day on integration is placed more or less on software engineering that involves integration of databases and models. A schematic of the organizational concept and module of an integrated information system should mirror much of what is available in current project management software systems. It would support all of the various project

professional views with their appropriate mapping tools, and all of the add-on modules would appear in the base system interface.

In summary, an integrated information system in my study should emphasize the following elements:

- Comprehensive information environment: Research emphasis is on expanding computer support to a GIS-based environment to the diversity of construction information management that is largely unsupported by current construction information tools. The system should encompass a comprehensive suite of information that supports a broad range of construction management functions. Also the overall system should be able to update constantly to include new topics or new information.
- Integrated data representation: A main characteristic of GIS-based integrated information system for construction management is that it provides an integrated data representation scheme that can be used as a guide for the development of a central repository of information about any functional construction task. The system, as a whole, should (a) contain a fairly complete and open model of the facility and the construction process, (b) enable dissimilar data to be easily integrated, and (c) deliver different views of the integrated data in a variety of formats.
- Intelligent function set: One of the primary asset of the GIS-based integrated system will be its ability to develop appropriate models and prototype to enable the evaluation of information changes by moving along different branches of an information tree. Case-based reasoning, artificial intelligence, and industry-specific function sets will be used to incorporate intelligence into the system.

- Flexibility: A GIS-based integrated information system offers a great deal of flexibility in the use of data, from a variety of retrieval options to diverse analysis and various reporting formats.

If these challenges are met, the construction community has much more incentive to adopt an integrated project information management system, because it would support a much wider spectrum of information and activities carried out by project management staff on a day-to-day basis.

Research Aims and Hypothesis

The objective of this research, which lends itself to the integrated system development goals described above, is to promote the awareness of GIS as a pragmatic solutions to the problems of integrity and consistency of information conveyed using a new GIS-based medium. The goal of this research targets efficient use of advanced GIS technology in the construction industry showing its potential to improve construction information integration. This research is intended to develop the framework for an integrated project information management system based on GIS technology and GIS based tools that can potentially enable project participants to access spatial analysis data that is site specific, to navigate and to manipulate information typically used in construction projects to support project management during various phases of a project and through the numerous changes that frequently occur. The primary objective of the research presented in this dissertation is to provide pragmatic GIS solutions to the problems of information integration in the construction industry. An important research

goal is to show how information integration in the construction industry can be improved by the use of GIS technology including model-based systems. A direct research objective is to utilize the GIS based integration strategy by building upon existing GIS software a prototype information integration system for the construction industry. The technical goal of this research is to provide a testing on GIS technology potential for construction information integration application, and to provide pragmatic solutions to the problems of integrity and consistency of the information conveyed using GIS-based medium, throughout the whole construction project life cycle.

The research has four primary aims to:

1. Assess the potential of GIS technology for information integration strategy,
2. Investigate the feasibility of establishing a GIS framework for integrating information in the construction industry,
3. Propose a methodology for the development of GIS-based integrated system.
4. Develop and test the GIS based integrated information system plan.

This research makes several assumptions and hypothesis about the GIS approach to construction information integration:

1. Methodologies developed by GIS for building an information framework and analysis purpose could be highly valuable for construction information integration.
2. In a GIS approach, the construction information can be managed in the integrated environment, thereby providing a consistent, coordinated, and well-represented

communicative platform to handle the complexity and vast amounts of information involved in construction project management.

3. GIS systems can be designed to be effective and efficient for a certain range of construction information integration tasks. Also it could be a medium that enables interactive procedures. Using a GIS as a tool in the information integration process may reach some significant new improvement.

4. From a modeling perspective, it is possible to build a model capable of providing support in the construction context. It is also assumed that such a model can be specified in sufficient detail to allow its embodiment in an intelligent GIS.

5. It is possible to build a GIS-based system capable of providing information integration support, which means:

- GIS will establish an information integration environment that enables all or most of construction information to be taken into the new system.
- There will be explicit representation of construction products and activities. The information relationship and context could be mapped into an information model.
- The construction project information integration goal can be achieved to some extent by the GIS-based integrated information system. The computational process can be formally defined and can be implemented.
- The system will provide a flexible system configuration to construction information integration task.

Research Tasks and Deliverables

This research is intended to develop the framework for an integrated project information management system based on GIS technology and GIS based tools that can potentially enable the project participants to access, navigate and manipulate information typically used in construction projects to support the decision making process during various phases of a project. This research attempts to construct, in general manner, a GIS based information integration strategy for concurrent construction project management. System integration strategy is developed based on GIS ability and construction project information management needs. The research starts from the study of characteristics of construction information. Contemporary GIS technology and its potential for using in construction information management then are identified. The information integration strategy of GIS-based system is studied. The information framework and system architecture for construction information management is proposed. An application protocol is developed building upon existing GIS software to testify the GIS-based integration strategy proposed.

The research is planned to occur in three phases. In each phase, the research progresses with tests that validate implementations produced within that phase of developments.

Phase 1: At this phase, the major task is to develop and test the general methodology of the research. This phase is aimed to set up a pilot system development plan. The detailed tasks include:

Test 1:

- Assessing characteristics of information in construction management based on the data sources, processing procedures and information classifications.
- Assessing existing information system based on data, domain and function.
- Assessing existing and potential information integration strategies based on previous step.
- Assessing the potential of GIS technology for fulfilling these strategies.
- Developing and testing the GIS based integrated information system plan.

Test 2:

- Identifying information sources and classifying the data type and format.
- Formatting and checking the raw data for construction project management.
- Developing and testing database management schema.

Test 3:

- Specifying of specific computing platform requirement for construction information management.
- Specifying the computational analysis function for information management.
- Developing and testing the data acquisition environment.
- Developing and testing the information retrieval schema.
- Developing and testing the overall system architecture framework.

Phase 2: Here, the research methodology will be refined using the test result and user feedback from Phase 1. This phase is aimed at identifying the operational problems and development issues. The detailed tasks of this phase can be categorized as the following:

Test 1:

- Analyzing the testing result and user feedback from test 1 in phase 1.
- Refining the GIS based integrated information system plan.
- Testing the GIS based integrated information system plan.
- Developing and testing the data acquisition environment.
- Developing and testing the information retrieval schema.

Test 2:

- Analyzing the testing result and user feedback from test 2 in phase 1.
- Refining and testing database management schema.
- Developing and testing conceptual modeling of data and their relationship.
- Developing and testing data processing mechanism.
- Developing and testing hierarchical, modular, and standardized database models.

Test 3:

- Analyzing the testing result and user feedback from test 3 in phase 1.
- Refining and testing system architecture framework.
- Developing and testing the project representation module.
- Developing and testing the information transaction module.
- Developing and testing functional analysis module.
- Developing and testing the graphical user-interface environment.

Phase 3: In this phase, the research product will be further refined. This phase is aimed at developing an application protocol and refining the research approaches. Tasks in this phase of research involve:

Test 1:

- Analyzing the testing result and user feedback from test 1 in phase 2.
- Refining the GIS based integrated information system plan.
- Refining and testing the data acquisition environment.
- Refining and testing the information retrieval schema.

Test 2:

- Analyzing the testing result and user feedback from test 2 in phase 2.
- Refining and testing database management schema.
- Refining and testing conceptual modeling of data and their relationship.
- Refining and testing data processing mechanism.
- Refining and testing hierarchical, modular, and standardized database models.

Test 3:

- Analyzing the testing result and user feedback from Test 3 in Phase 2.
- Refining and testing system architecture framework.
- Refining and testing the project representation module.
- Refining and testing the information transaction module.
- Refining and testing functional analysis module.
- Refining and testing the graphical user-interface environment.

The major deliverable of the research is a GIS solution for basic integration needs, including detailed interpretation of the literature, step-by-step evolution of analysis, and a technical sound model of the GIS based construction project information integration system. Additional deliverables provide novel solutions to the coordination of multiple project information sources, such as project models defined in a conceptual information

framework, and flexible data management and integration mechanism. With these deliverables in mind, the prototype system has been generated to testify this technology. This development is based on a set of theoretical models, which define the concepts, and strategy of integrated information system. Based on theoretical model, a set of computational model defines the representation scheme for the information structure and information processing mechanisms. A computational prototype is developed to serve as a test bed to demonstrate and verify the effectiveness of the theoretical model. The result of this research includes the project information modeling framework, the system development plan and the implementation of the modeling framework and the prototype system. A prototype GIS-based integrated information system, called Construction GIS, is developed based on the strategy and approach proposed in this research. The final product provides data structures, user interface components, and output mechanisms that effectively connect the empirical analysis models to standard GIS. The system has been put into case testing, and feedback is used for refining the proposed research methodology.

CHAPTER 2 MATERIAL AND METHODS

The philosophy behind GIS-based integrated information system development is to construct a model in terms of a GIS system that matches the construction project description as closely as possible. The design of a model for the provision and interchange of information within a project implies that the real-life project can be modeled in a way that allows functionality in a system. The information model should have general usefulness for performing different functions, capture significant levels of detail, include a wide breath of project information, and have flexibility in what information can be represented and how the information can be used. It will be necessary to understand and model the object content that satisfies the requirements. This means that the information model representing all the information about a building project requires considerable inputs of logical and practical analysis in its design.

The information framework is designed to address the information needs and also attempts to support the integration of the information required using the GIS-based product and process representation schema. Therefore, for the information framework to make significant progress it is essential to integrate the potential of the GIS model and research results for construction project modeling. As a result, the information model is hybrid in a sense that the information framework aims to address the modeling issues by building on the basic model of GIS, while according with the findings of the construction

product modeling. The geo-relational approach is superimposed on project modeling methodology to achieve a comprehensive information framework that supports a more coordinated and effective communication within the information integration process. Thus, we examine, how a geo-relational model contributes to meet the information integration modeling challenge. Then, we emphasize how to extend the geo-relational model to fulfill the requirements of the special purpose of project modeling using the object-centered methodology.

Project Modeling

Previous research efforts have addressed appropriate bodies of construction project information modeling. To date, it appears that most of the fundamental concepts about description of construction activity, construction products, processes, resources, participants, etc., can be referenced from previous research effort. While these models are useful for definition of data and product, the conceptual model proposed in this research can employ one or more of these models in various stages. The integrated data modeling technique extensively uses available data modeling techniques for the purpose of data analysis. Its main approaches are identification of entities, keys, and attributes from data sources associated with each system component; and establishment of entity relationships, and justification of the functionality of these relationships, based on semantic description of the project management schema.

A project model to address the information integration issue is developed from a review of the work others have performed in the area of classification and collection of construction information. Project modeling research works, such as ISO STEP and IAI

IFCs, have influenced the project model proposed in this research. The proposed project model adopts most of the fundamental concepts required to describe construction information, which provide extensive and attractive features that are suitable to the development of information frameworks. These concepts include formal representation of entities and relationships, integrated frameworks, spatial and non-spatial data integration, as well as multiple view and dynamic models, all of which are discussed in greater detail in the following paragraphs.

A basic modeling approach is to define the entities and relationships in the application domain. Although each construction system is unique, the operating processes of its component resources are usually generic. The basic concept of a construction project can be extracted for the definition of project model, as follows. First, a top-down analysis identifies the processes or functionality involved in the domain to produce an application process model. This model leads to the domain data requirements by listing the input and output information flows from each domain process. Second, a bottom-up analysis explores the typical documents used on construction projects, and identifies their information content to give a secondary means of identifying the domain data requirements. From these two approaches, the data requirements are listed first as a list of the basic topics, then as a comprehensive list of information topics required from the data model. Each of these topics requires specific object or entity definitions to define in a data model. These entity definitions, including entity attributes and relationships, are designed to produce a preliminary data model. This information is then classified with standard procedures and represented in an information model. The collection of all of the data topics makes up the scope of the project model.

An integrated data model allows better presentation for major project information and gives a variety of interested parties a simple tool to examine project conditions. Common models of all project information (product, process, cost, organization, resource, etc.) are required to completely detail all aspects of a project throughout its life cycle. Furthermore, the information relating to virtually all phases of project management can be tracked within a computer system, requiring at least some, if not all, information items about the information to be structured in the data model. This requires rich and flexible formulations of project models for building form and systems, activities, resources, organization entities, documents, etc., along with the ability to create associations among such model partitions. A properly implemented model can provide the glue that binds most corporate information together and produce an environment that is highly conducive to improve productivity. This approach is intended to support research in all the elements of the project life cycle. To achieve this objective, an information management framework that reflects the project handles the stage-by-stage information management in a given project cycle.

Spatial data is commonplace in many construction applications. Present research projects deal with the project modeling and the development of integral and flexible data exchange models emphasize the importance of spatial data representation. Most construction operating functions require identification and location of many components comprising facility, configuration and operating parameters. Construction personnel have historically applied spatial or location information to assist in managing their data. For example, construction depends on CADD systems to produce the engineering drawings needed to construct and maintain the systems, and finance depends on facilities-related

systems to identify the physical assets and permit proper recording and costing activities. The project model proposed in this research adopts the approach of representing spatial data in a unified way to facilitate information communication between professionals. Also, it emphasizes the dynamic transformation of the spatial data into meaningful objects, to meet specific needs of different applications. It also recognizes the fundamental and important interrelations between the spatial and non-spatial attributes of facility components. The representation and linkage of these two types of information using computer-aided design/engineering (CAD/CAE) systems have received considerable attention during the past decade. In the research project model, the formal representation of spatial and non-spatial attributes of physical objects constitutes the methodology for defining and accessing the project information. Spatial information is modeled by specialized geometric representation schemes and often shared across disciplines; while non-spatial information is captured as attribute-value pairs in taxonomies of attribute classes that are discipline-specific.

Information analysis depends on the perspective of the viewer. Each participant in a project has his own view of the information about a constructed facility. In a construction project, various data types have different semantics associated with them. Project data are perceived differently by different people. Thus, one obtains numerous interpretations of the same data. The research results from previous work in project modeling that emphasizes the importance of multiple view modeling. The key to good information is the ability to detect and bring together all the data that applies to the issue that the user is concerned with. Some of the activities can be concealed behind the component in place. The modeling approaches adopted by this research utilizes semantics

that reflect activities identified during view definition. The research result is consisted of the product model that does not depend on the specific participant's view. Also included is a process model that uses several methods to retrieve and update the project information from product model according to a given participant's requirements. The view-based approach to identifying semantics also tends to identify perspectives at a relatively small grain. Rich semantics of the project view representation is thus permitted. This facilitates context management and persistent support to guide users to the useful information pieces.

Dynamic information modeling is another main approach adopted in this research. One of the clear deficiencies of any construction project data is that it rapidly becomes out of date. When communication involves a static model, additional exchange is required. A dynamic information model would automate this exchange and, ideally, provide active notification for the change to the interested parties. Modeling methods for linking these data sets, along with transition related data generated clearly have an important role to play. Also, if data updating are conducted with the addition of a time dimension, then conditions of each work activity, areas and elements can be tracked over time. This monitoring capability allows a project manager the ability to see if certain procedures are performed as expected and allows these designs and procedures to be modified as necessary. Automating coordination activities will be as important as information sharing. Explicit representation of relations and dependencies between design elements will support the coordination goals.

In general, project modeling provides a fundamental paradigm that supports:

- Comprehensive information base by including relevant engineering knowledge and project life cycle information.
- Integrated representation of spatial and non-spatial project data.
- Multiple views for various engineering tasks.
- Dynamic model for persistent and active coordination and control.

The accurate modeling of construction operations requires large complex models, which are difficult to develop and validate. A need to simplify models has been identified. This has resulted in reasonably solid constructs for abstract concepts such as physical object, representation and identification. This is used as the basis to develop atomic models and stored in a model library. Atomic models from the library can then be surveyed by project-associated data and modified to form computational models. The environment will then identify the appropriate linking structures and assemble the working model. These models model the information integration process and are combined to create a system model.

Geo-relational Model

This research is intended for using GIS technology for construction project information integration. And it implies that the geo-relational model approach is the main modeling approach adopted for this research. GIS is frequently referred to as geographically oriented computer technology, with integrated systems that permit the analysis, acquisition, management, and display of different types of spatial related information in a single integrated information model. There are now a large number of

good introductory texts on GIS and geo-relational modeling that presents a wealth of alternative definitions. A very brief definition and description of geo-relational model is that it deals with spatial data and linking the non-spatial attribute to the spatial entity in terms of the organization of data.

The optimistic view of geo-relational modeling is that it has the potential to have great utility in the construction project information management. The construction industry is ideally suited to application of geo-relational models as its operation is predominately based on physical assets covering a certain spatial area. The feature of geo-relational modeling that could be applied to the modeling framework for this research includes the general focus on spatial entities and relationships, linkage of non-spatial or attribute data to spatial information, and geo-reference integration. These features are discussed in greater detail in the followings.

Geo-relational models serve distinctive needs when spatial descriptions play a central role in the observation. A geo-relational model is optimized for spatial data. It stores information in data structure appropriate for spatial data that might not be supported in the relational model. A feature is the principal data object in the formal representation paradigm of geo-relational model. At the pragmatic level, the world as perceived by geo-relational models is composed of features. A generic definition of feature is any spatial entity with some semantic content necessary to a particular context of reasoning. The feature attributes are categorized into two disjoint but related types: spatial and non-spatial. Spatial data have the physical dimensions and provide the reference to which non-spatial elements are automatically linked. Non-spatial attribute data may well include the parameters that reflect descriptive measurement. In this

research, the project model uses spatial data as the basic medium for information integration, while spatial information pertains to geometric attributes and topological relations of physical entities in a project and on-spatial information describes all other characteristics of physical entities, such as functionality and material properties. The geo-relational model links apparently disparate data sets together by location. Thus our model enables simultaneously accessing, updating and processing the spatial and non-spatial data associated with each geographic feature.

In a geo-relational model, all attribute information within the model is associated with one or more spatial features, even though it may require a chain of joining operations between numerous tables in order to establish this relationship. Attribute information is associated with point, line, zonal or other spatial entities that describe features occurring in the real world. In this approach to data integration, spatial entities are usually linked with their associated attribute data by means of a common spatial key (commonly a unique identifier or ID assigned to each spatial feature). By tightly integrating such entities, referential integrity can be assured. The result is holistic: attribute information can be found by selecting spatial features, and spatial features can be found by querying attribute information. This approach allows the project model developed in this research to create and manage the spatial information on the feature type as well as the descriptive data on any other kind of characteristics associated with each feature. This also gives the user means to query and analyze variation in an interactive fashion.

Most geo-relational models usually adopt a dual data storage strategy with spatial data held separately from the attribute data. The system also maintains links between

software modules that handle each information type. Spatial data are typically represented using a topological spatial data model, and thematic data are usually stored in the tables of a standard relational database. Different sets of attribute information are stored in different attribute tables, and the relevant information for a given set of spatial features are accumulated by relating (or joining) two or more tables of information. The collation of information from the attribute tables may be carried out in several ways: for example, by exact, hierarchical, or fuzzy matching.

This dual database and model is enhanced through open data structure techniques. The geo-relational model is very flexible in the types and structures of data it can receive and integrate with other data. The geo-relational model provides the ability to not only attach alpha-numeric data as an attribute to any element, but photographs or video can be attached as a specific attribute of any element in the geo-relational model. Diverse types of data are combined and integrated into geo-relational model in the form of a database, program or system to hold data conveniently, for use in a variety of ways. By doing so, the project model in this research accepts data from multiple sources, which can be in a variety of formats.

Additionally, a geo-relational model, apart from combining different formats of data, links data together based on location. One attribute of most GIS approaches is the use of geo-references as the primary means of storing and accessing information. The base of a GIS database is a uniform spatial referencing system for the data in the system, which also facilitates the linking of data within the system with other data. These different types of data are organized into a special-purpose digital database in which a common spatial coordinate system is the primary means of reference. Once the data is

geo-referenced, the GIS will search through all of its stored information to retrieve all recorded information associated with that geographical feature.

Conversely, the entire databank can be searched to find all attributes associated with a certain spatial reference point. The value of these measurements are not fixed, but may be selectively adjusted to produce alternative representations, or combined to produce new information that is not a part of the input data. It provides a better way of managing assets and assists in providing the relationship between pieces of information while maintaining their individuality. At the technical level, a key feature of the geo-relational model is related to its ability to perform a map into a logical entity whose elements are subjects to various combinatorial operations with other sub-sets of information.

Once the basic data layers are created, the analytical functions of the GIS can generate automatically many of the variables required. It arranges information about different set of issues as a set of thematic layers with each layer displaying information about one characteristic of the region.

Each of these separate thematic maps is referenced to the grid of a location reference system to which all the maps have been precisely registered. Information displayed on the different layers can be compared and analyzed in combination. By doing so, geo-relational models provide the flexibility to consider the relationship between different sets of information. Information from two or more layers might be combined and then transformed into new layers insofar as it involves adding and subtracting information. This is done through introducing a series of data layers in the database. The ability to separate information in layers, and then combine it with other layers of

information is the reason why geo-relational models hold such great potential as a powerful information model for information integration.

In summary, geo-relational models offer several contributions to the data representation of a physical object in a construction project by providing:

- Spatial data modeling of the project and using it as the basis to access other data,
- Computer-based representation of spatial and non-spatial attributes of physical objects in data structure appropriate for spatial and non-spatial data,
- Linking between spatial and non-spatial data,
- Association of different items of attributes through the shared spatial entity, and
- Optimizing the organization and structure of the database for dynamic rendering and navigation at multiple levels.

Object-centered Modeling

An information model that is capable of dealing with the complex characteristics of construction information outlined in previous chapter requires having a set of specialized types of engineering information, such as analysis models and algorithms, constraints, time-dependent processes, and many other types of data that are somewhat unique to the construction engineering domain. Because of the complexity of construction information and the flexibility in object definition and the support of multiple levels of abstraction in construction, an object-centered project model is

recommended by many researchers (Chin, Stumpf and Liu, 1996). This object-centered methodology is capable of representing the complex information in the construction industry. It also provides the means of showing the complexity of the problems in construction project process and the engineering entities involved in models, which are based on real-world concepts.

This research takes advantage of the rich semantics of data by providing an object-centered model. The notion of semantics is central to the modeling methodology. Semantics can be thought of as an object-centered model describing the information needed to support a single well-defined activity. One of the key ideas in the research model is to provide a semantic modeling method by which a semantic representation of a construction project can be mapped. Semantically, the object-centered technique can be exploited for project data definition and integration in the modeling. By predefining the object, a description of the project model could be attached to the feature.

Another advantage of using object-centered technology is its nesting capability of breaking the property sets to support different requirements of information details at different stages and levels. The project sketch's definition then is an n-array tree, the root of which contains the minimum information of a construction project. The sub-trees associated with the root represent the structure of the product's components. To descend a level in a branch of a tree implies that a decision has been made or that information has been gained regarding the components of a part of the product. The existence of an internal hierarchy of objects and details allows an evolutionary modeling approach. A fine granularity of product knowledge representation is thus permitted that facilitates a

hierarchical decomposition approach. The hierarchies allow decomposition to take advantage of the object-oriented concepts for specialization and inheritance.

Extensions of the object definition also incorporate automating coordination activities to automate data dependencies between elements and ideally provide notification for the change of the interested parties. The underlying concept is that there are certain data that might get into an active state in a project, and consequently influence certain items and cause changes. The point of building active functionality like this into the database system itself is to ensure consistent usage and high performance. This approach is superior to passive database for enforcing general integrity constraints and enabling triggers, as well as for supporting data-intensive expert systems and workflow management applications. The active nature of the object-centered modeling approach permits the system to constantly monitor data relationships and integrity.

It is important to explore general-applicable characteristics of information modeling and management concerns, the modeling mechanisms include aggregation and composition, multiple view representations of objects, relationship and integrity constraint, etc. The object-centered modeling approach provides a systematic approach for long-term information system development by relating pre-defined tasks, control devices, and peripheral devices into real-world, event-driven, and multi-tasking object definitions. Several contributions are thus realized:

1. Representing common information requirements for all applications with a single set of project object definitions described by a computer language.
2. Mapping the semantic contexts and properties of real-world building project objects and their relationships to a computer language format. Therefore, a wall is

represented in a computer as an object (i.e., not a series of lines) that contains semantic attribute values for the wall, such as its thickness.

3. Providing data structures so that they can be instantiated at run-time and information exposed by the instances' interfaces can be shared and exchanged among different computer applications for different industry domains.

4. Supporting live objects that develop dynamically over time (i.e. the object's state changes), throughout the entire life cycle of a building project.

5. Supporting semantic expressiveness of data models that allows generalization, aggregation, interaction, inheritance and structural relationships.

The major challenge of proposed modeling approach is its complex complexity. The research is intended to add object-oriented facilities to existing geo-relational model in GIS to implement project modeling. A framework has been developed for modeling and communicating information about construction project using three components: a geo-relational modeling kernel, a multi-layered structure of underlying project modeling base, and the object-based procedural interface. The geo-relational modeling kernel contains low-level specialized spatial and non-spatial data structures and functions that are useful for implementing the detailed knowledge of the particular project non-manifold modeling schema. The object-based design and procedures join the links between these two layers by providing a high degree of modularity and encapsulation, resulting in an open and extensible modeling framework.

CHAPTER 3

RESULTS

The GIS-based information integration system has been developed in three stages. System development started with the construction of an object-centered, GIS-based geo-relational modeling framework based on a systematic approach. It is considered as the hub of the proposed methodology through which information integration is achieved. The next stage involves system implementation that supports information requirements of the information integration process using appropriate system architecture. The specification of the system approach along with the mechanism allowing the integration means is incorporated into the GIS system. The third stage assembles the modeling framework and system approach into a prototype system called Construction GIS, which is developed and tested. Feedback from test procedures is used to refine the proposed methodology.

Project Information Modeling Development

Fundamental to this work is the development of an integrated information framework or project model that represents the information requirements for construction project information management. The purpose of the project information modeling development is to represent standard construction project database structure. The elemental proposition of a project model are extracted, developed, and analyzed from

various literature sources, and the result establishes declarative knowledge. Because there is no unified or standard method to decompose a complex domain into elemental propositions, the propositions developed and analyzed might be different from one project to another. However, the goal of developing elemental propositions is the same; that is, to define facts about the domain that cannot be decomposed into subordinate facts, while preserving the original meaning and purpose. This step represents the construction data and knowledge in a form of templates that can be used for database modeling. The main technique is to identify the functional requirements for the data model, then to devise the data topics required to meet these functional requirements, and finally to fully detail object definitions that describe each data topic. Functional analysis consists of developing a hierarchical listing of functional capabilities the data model would support. Functional analysis also identifies functional categories/topics that the system and thus the project model should be capable of working with. The declarative knowledge involved in project modeling development proposed in this research can be categorized as:

- Project feature object modeling, and
- Project view object modeling,

Project Feature Object Modeling

Relying on a GIS-based information model and object-centered modeling methodology, a feature object is the key concept for data representation of construction project information. In this model, feature definition in a geo-relational model is extended

to objects. The term object is used in a conceptual, generic sense to denote an entity characterized by attributes that have associated values. Objects are created to contain all non-geometric attributes as well as a link to a unique geometric entity. That is, a feature's non-spatial attributes in the product model will be treated as a set of objects such having a set of attributes. The object class contains the feature description as attributes and it refers to its primitives. By applying this approach to a GIS system, semantics can be added to the data model. This means that, as soon as an object is attached to the feature, all related attributes of this object would be implicitly available.

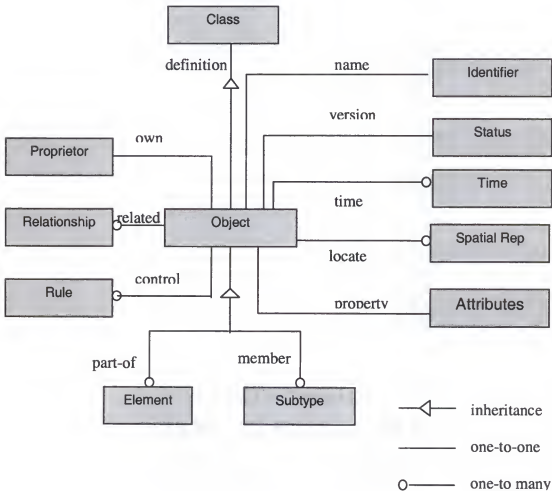


Figure 3-1: Feature object definition

A feature object is formed by assigning or deriving semantic description and action rules of the project component in terms of feature objects. Basic project feature object classes are defined by creating libraries of feature object classes. For each class, we define attributes of the objects, their behavior, and the relationships that this class of objects can have with other objects. Common attributes and behavior of objects are extracted at a higher level for improved reusability and interoperability of project objects.

After relationships between objects are defined in each object class, the project information objects are integrated into a global framework. All project object instances in the project central database are created using the templates defined per their respective classes. When a feature object in a configuration is created, the database management system knows which attributes/fields of the object should be there and the associated objects should also be automatically created. For any feature object, the following property can be described: a unique object identifier, a proprietor, database status, time stamp, spatial representative, attribute classifications, hierarchical structure, relationships and rules controlling objects, as illustrated in Figure 3-1.

Identifier

Every feature object has a unique identification name of its object class. The instance objects receive names selected by the user, which is the most generic element information and is stored directly within a physical entity class. It is this identifier that is used to uniquely locate any feature object instance through the database management system (DBMS) mechanism, the query. Most importantly, this identifier to some extent is representing the whole definition of the object and all the instance values with the

instance object. The object itself and the attribute value can then be retrieved through this name. For example, 'wall_1' refers to the component representing the building feature with the attribute of material, quantity, measure and so on. It also means the pointer to the attributes associated with object. In practice, an identifier is created by associating any object definition in the object library with a string name.

Proprietor

Each object will have a proprietor, which is either a user or a business unit. The proprietor will have the privilege of modifying an object item. The proprietor implements context integrity constraint in the database to restrict access to the database and limit the changes that a user or application can make to the database. It provides support for the actor roles and rights during the entire project life cycle. A role is granted specific rights (read, write, approve or delete) for both attributes (for read, write and approve rights) and object instances (in particular for approve and delete rights). The rights defined on an instance are applicable to all its attributes if not defined on individual attributes.

Status

The feature object is mapped in a structured environment. The primary reasons for providing a structured environment in which such changes are recorded are firstly to provide a complete project history, and secondly to facilitate backtracking. In addition, a structured historical versioning environment reduces ambiguity by standardizing version numbers and allowing examination of previous versions of objects. An environment in which the concept of proposed versions is supported, and in which proposed versions are

distinguished from historical versions, improves project information reliability and communication. The feature object uses status to control different versions. The status of a feature object keeps track of the changes enforced on the feature object and locates the most up-to-date version. Status is set according to different states of the object. In general, there are two status values for different versions: "E" and "N". The most up-to-date version has the status of "N" while all the old versions have the status of "E". When a new version of a feature object is entered into feature object database table, the previous most up-to-date version becomes the old version and the status value for that version is automatically changed along with the new version get the most up-to-date status.

Time stamp

For maintaining the data version and database history, a database time stamp is needed to represent the time of data input. Each feature object has a sequence of states representing the knowledge for a given period of time. The time point, when database obtained changes to the state of the object, marks the beginning of the valid time for the database records describing the new state of the object. Another time dimension independent of the valid time is the snapshot time, which is the time when a snapshot of the state was taken. When the exact valid time cannot be recorded, the snapshot time can be used to maintain information and interpret project history.

Spatial representation

The spatial representation forms the basis for the information reference. All attribute information within the model is associated with one or more spatial features.

Spatial information of a feature object is modeled by specialized geometric representation schemes and often shared across disciplines; while non-spatial information is captured as an object that defines attribute values.

Maps/drawings contain the spatial entity that is extracted from three-dimensional CAD elements. Spatial representation of a project typically consists of a set of graphic entities related to one another through a Euclidean coordinate system. The spatial representation of a feature presents a high-level geometric description of that feature. This representation is used primarily for reasoning about the topological relations of that feature with other features. Spatial representations can be categorized as geometric or topological. Geometric attributes can be used to specify traditional CADD attributes of location and dimensionality. Geometric shapes of possibly mixed dimensionalities receive various geometric information. Topology is developed specifically for supporting the referential spatial representation scheme proposed in GIS.

A simple spatial representation is a single, connected, dimensionally homogeneous geometric element, i.e., point, line, and polygon (area). Based upon the dimensional axis, these entities are grouped as pointil, lineal, or areal for zero, one or two spatial dimensions, respectively. A compound spatial entity representation contains the semantic relationship and semantic grouping of the spatial entities as necessary to relate to the information management task. Also, there is the need to convert 3D coordinate information into 1D spatial keys that can be stored as semantics, which can then be indexed in the normal way and used for fast retrieval of spatial elements.

Spatial entities can locate, topologically associate, identify, or represent world entities on a drawing or map. They constitute respectively three classes of operations, i.e.,

local, focal and zonal. The local operations compute new values for each location on a layer as a function of existing data explicitly associated with that location. The data to be processed by these operations may include the zonal values associated with each location on one or more layers. Local operations include classification, generalization and overlay operations. Focal operations, including neighborhood operations and connectivity operations, compute new values for every location as a function of its neighborhood, which has a topological and/or directional relationship to a particular location. Zonal operations, including search operations and measurement operations, compute new values for each location as a function of existing values associated with a zone containing that location. Table 3-1 summarizes the basic classes of spatial operations and provides representative examples.

Table 3-1: Basic classes of spatial data operations

Local Operation	Classification	assignment of new attribute values to individual locations on a layer
	Generalization	reduction of detail associated with individual locations on a layer
	Overlay	assignment of new attribute values to individual location resulting from the combination of two or more layers
Focal Operation	Neighborhood	assignment of new attribute values to depict their distance, topology, or direction in a neighborhood with respect to the focus
	Connectivity	assignment of new attribute values considering the values associated with locations in the immediate or extended vicinity
Zonal Operation	Search operation	retrieval of information characterizing individual locations on a layer that coincide with zones of another layer
	Measurement	assignment of new attribute values to individual locations on a layer that correspond to a measurement (e.g., area, length) characterizing their zones.

Spatial representation is complex and dynamic rather than simple and static. Feature objects allow spatial representation to have multiple meanings within different interpretations. For one feature object, different views may use different shape descriptions (e.g. topologies and dimensionalities) to express the spatial information about components that are present in a functional view. While each feature object has one primary representation, which is also accessible across a different set of views, it may have several secondary representations, which is accessible across different views. The primary spatial representation of a feature presents a high-level geometric description of that feature and provides the feature's identity. The secondary spatial representation is used mainly for reasoning about the topological relations of that feature with other feature in a particular view. Secondary spatial representations resolving to the same primary representation correspond to an identical spatial entity.

Non-spatial attributes

The structure of a feature object semantic description is defined by a collection of characteristic attributes. The attribute represents the atomic element for modeling the non-spatial properties of facility components. Attributes are non-spatial information describing characteristics of feature object, such as functionality and material properties. Dividing the attributes into several attribute sets allows the display of only the needed attributes for each step. It is less confusing and overwhelming than an object showing the whole list of attributes. An attribute set is a fixed grouping of attributes, the purpose of which is to bundle attributes that are logically grouped or associated. The attributes of an attribute set are intended to describe the same property context. For example, for

describing the shape of a feature object a number of attributes are needed: length, width and height as a minimum.

The feature object attributes are managed by relational database tables. The attributes are mapped onto tuples of the appropriate tables that correspond to the object definition. The mapping from the non-spatial data abstractions of feature object to the underlying RDBMS is achieved by a set of special procedures. The functional core interface in turn encapsulates the data modeling primitives and access mechanisms specific to the underlying relational database management system (RDBMS). It thus provides the basis for implementing the operations for manipulating the high-level functional abstract attribute type of the information model.

The feature object database table remains an empty structure or template until values are assigned to its attributes. These tables typically contain attributes in the form of fields. An attribute field has a specified data type which defines the domain of its values. When an object is associated with a particular attribute sets, the attributes of that object and its super classes (if any) are instantiated with either default attribute values or methods for computing certain attribute values. Attributes are set to 'null' if no values are specified for them. The domain of an attribute specifies the type and range of values that attribute may have, such as integers, real numbers, or character strings. This supports the enforcement of valid values for feature object attributes. The database accepts and manages restrictions on attribute values and relations to guarantee basic well-formedness of the data. On the other hand, it is not required that the database enforces well-formedness in the physical terms, i.e., the database tables could have not all the attributes and records for each attribute.

Each feature object defines certain fundamental attributes associated with it. Also, an alternate mechanism is provided for dynamically defining attributes for either individual occurrence of an entity or for a group of similar occurrence. This is useful when there are properties that are common to many occurrences of a particular entity. These properties may have the same value for all occurrences of the defined type, or they may have different values for each occurrence. In summary, property type sets are dynamically defined properties that can be either extended entity definitions, type-defined properties shared among multiple occurrence of an entity, or type-defined properties whose value are specific to single occurrence of an entity.

Hierarchical structure

Hierarchical structures are defined for feature objects. Three hierarchical structures are very important: a class hierarchy that defines how the super-level class patterns the sub-level class, an aggregation hierarchy that defines how objects are made up of sub-parts, and a specialization hierarchy that defines how an object can be generalization of several more specific sub-types.

There is class hierarchy among different feature objects. Every feature object class is in a certain level that has different influences on other feature objects. The first level class has the power to influence the initiated second level class object and then the third level class through the second level class object. The overall definition of a feature object library progresses from the higher level class objects to lower level class objects. As showed in Figure 3-2, the product feature object is the first level class, which means it initiates the definition of second level feature object class such as activity object. The

activity feature object in turn initiates a method feature object and a resource feature object. The product feature object also could initiate the resource feature object. And the activity feature object and resource feature object initiates the participant feature object.

This process of defining a feature object complies with the real world project process. For example, if a change order is issued to change the initial design, the change order information will affect the component directly, the corresponding activity will need to be reconsidered, and resource information might be changed.

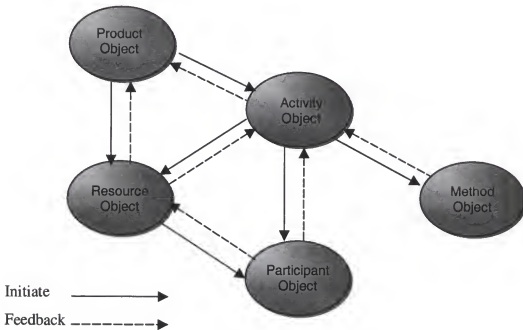


Figure 3-2: Feature object classes' relationship.

An aggregation and specification hierarchy within the feature object list would allow the description of the project to be interactively refined to increasingly greater

levels of detail as additional information become available. Also it will allow the evolution of a project model as the project progresses.

The aggregation feature object is called the target and the components are called parts. A flexible composition mechanism allowing for multiple levels of aggregation has been provided in feature object representation. Composition is the aggregation of heterogeneous feature object to define another feature object. There may be multiple compositions describing the same target, for example one set of finite elements to define the shape and another to define the structural performance. Sub-components, being feature objects, can be decomposed to allow any number of decomposition levels to exist within a component, as illustrated in Figure 3-3. The same part may be in multiple compositions. Compositions may be defined in top-down or bottom-up fashion.

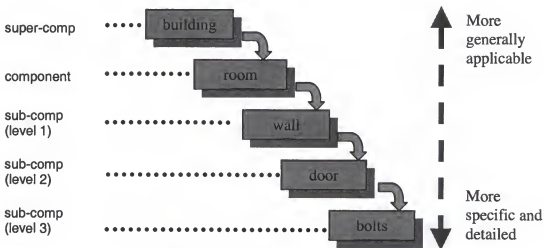


Figure 3-3: Aggregation of feature objects.

Similarly, the system contains descriptions of all types of feature objects generally used within the class of construction, arranged in specialization hierarchies. If multiple sub-types exist for the target's feature object type, then these represent the alternative techniques that can be used for carrying out the abstraction.

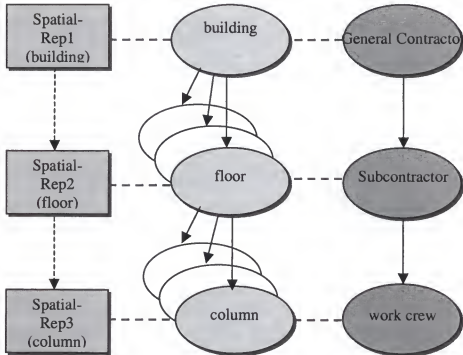


Figure 3-4: Hierarchical structure mapping.

The subtypes/subparts inherit the properties of the super-type/super-part besides their own properties, as illustrated in Figure 3-4. The mapping involves a process of adopting a high-level “seed”. That is, the attributes, relationship and constraints are propagated to sub-types and sub-parts. Before the data description of the sub-part/sub-type can be stored in the database, it is first validated to ensure that it is a true subset of

the super target. All feature object definitions in the sub-part/sub-type are matched with the definitions in the super object. The cardinality of each attribute in the sub-part/sub-type is checked against the super object definition. The upper and lower bounds in the sub-schema must lie between the upper and lower bounds of the corresponding object attribute. Once the sub-part/sub-type has been parsed and converted into an object representation, it is stored for later use as part of the data of its object. By doing so, the resolution levels of component representation can be facilitated.

Relationship

Relationships are an important part of the knowledge represented in a database. Explicit representation of relations and dependencies between elements support coordination goals. Three types of relationships between feature objects are defined: hierarchical, functional, and spatial relationship, which may be one to one, one to many, or many to many.

A hierarchical relationship, also called inheritance relationship is provided by a parent-child relationship between objects. The inheritance relationship exists between super-classes and subclasses as well as super-parts and sub-parts, super-types and subtypes. Since a parent object is a part of a specific type object, the inheritance relationship of the model structure can propagate downwards. Multiple inheritance relationship is supported. However, early on we determined that the database model would support only single inheritance, because the anomalies and ambiguities inherent in

multiple inheritances cannot be resolved consistently across different programming languages and object-based representations.

Functional relationships can exist between any two feature objects in the system, and are only limited by the semantic structure of the domain knowledge instead of the implementation environment. The functional relation results from the technical relationship between the functional systems. A functional relation describes whether an exception (e.g., design change, work delayed) generated from one feature object will have any impact on the work of another object. These functional relationships are captured through a series of analyses about requirements and solutions of each feature object class. The clarity of these relationships depends on their semantics, which are defined within the domain. Two types of functional relationship should be distinguished – technological and organizational. Technological dependencies may be derived from technical conditions of feature object. For example, the erection of a partition in a space requires laying down of a floor surface in the same space (regardless of the steps performed), or wallpapering that requires the completion of partitions (regardless of their composition) etc. Plastering a vertical masonry shaft may require the completion of masonry activity on all floors adjacent to the shaft. The organizational dependencies between feature objects may be derived from the organizational condition of feature objects. It applies to the situation that deleting or adding one feature object requires the deleting or adding of another feature object. For example, creating of activity objects requires the creating of material, labor and equipment objects in specific domain.

The spatial relation depicts the relationship of (a) spatial support (for example, a component to be installed in a subsequent activity is placed upon a component installed

in the preceding activity, e.g., wall is placed upon a floor slab), (b) proximity (for example, a component to be installed in a subsequent activity is placed next to a component installed in the preceding activity, e.g. a backfill is placed next to a wall) or (c) coverage (for example, a component to be installed in the subsequent activity covers a component installed in the preceding activity, e.g. a reinforcing steel which is embedded in a concrete). The spatial relationship may be derived from the conditions of feature objects in the same space or in adjacent spaces whichever is appropriate. Same as spatial operation mentioned in previous section, spatial relationships are defined as local, focal and zonal.

The majority of the relations are expected to be "complex" relations. That is, attributes and additional information are attached to the relations themselves in addition to the related objects. In order to implement complex relations, additional classes would be defined to represent the relations themselves. The relationship object defines the attachment of objects to each other as required by specification or optional. The representation of relationship objects centers upon the use of reference links to related entities. Each relationship contains references to the related object as well as a definition of the type of relationship between the two objects. In this centralized representation methodology, modifications to relationship definitions can be implemented without disrupting the underlying representation of the physical entities. Through this implementation, a single method to determine all relationships can be stored in a single location rather than being dispersed over several object hierarchies. In this centralized representational methodology, modifications to relationship definitions can be implemented without disrupting the underlying representation of the associated physical

entities. In a database, relationship sets describing many-to-many relationships between objects are also represented by a table or data value. The tables contain columns that reference the objects being related, together with the definition of the relationship itself. Since a relationship between entities is directly represented as a table, there is no requirement for explicitly creating pointers or linkages between data records.

Rules

The active nature of the construction feature objects requires the system to constantly monitor the results of any changes. A rule-based framework is added as a means to help enforce data integrity on database during interactive updates. All changes to object instances are handled through the use of rule supporting events for flexibility in working with runtime-dependent constraints on changes to a given feature. Rules are defined at the point the object is initialized. The rules for establishing the relationships and computing the updated values are stored in the respective classes. The rules can act on both object attributes and object.

The rule is a simplified structure of the variables and interactions that influence the database being analyzed. These rules incorporate a relatively simple control system for regulating database dynamic behavior. Rules in this framework can be defined to “fire” upon occurrence of a particular event, subject to arbitrary conditions anywhere in the database. Should one of the rules be triggered and its associated conditions hold true, then a predefined action would be carried out. The rule is also used for communicating updates among feature objects. When an object is created, associated objects instance should also be automatically created. If linked objects are to be created, the cardinality of

the relationship determines the number of objects that must be created. Changes to a particular object can facilitate other objects in the instantiated model.

The basic idea is that any rules having events defined for the current operation will have the opportunity to check for any particular conditions in the database that are of interest. Also it would be desirable for an object to be able to notify the associated object of changes once the value of itself or its attributes has been modified. Once object received the notification from system, the value of affected attributes will be updated automatically. When a rule fires, the feature object can communicate with the associated object in three ways: by broadcasting a change event with a message protocol; by indicating that the object has been changed; and by requesting that another object be allowed to make a change. After all rules' conditions have been checked, those which evaluated true are sorted in priority order, and their respective actions are performed.

A rule carries the following information:

- A set of pre-condition constraints that define the rules required for a feature object to be executed,
- A set of post condition constraints, identifying the rules satisfied
- A set of actions associated with objects.

The logic implemented within the rule is that if the pre-condition constraints for an operation become false, then the post-condition constraints that the operation satisfied are also no longer guaranteed and must be set to null. This can result in a forward chain, setting to nullify all data derived from the changed values. The object packages must have the ability to ensure that state changes are made in a controlled fashion, i.e. in a consistent state at the end of a state change. This implies satisfaction of pre- and post-

conditions with respect to the object itself and other objects to which it is related. The pre and post conditions to be applied may change with changes in the state of the object itself and possibly in state changes in feature objects related to it. In the project model, rules and support methods basically perform similar actions: creating/deleting objects and assigning/modifying object values. The key methods include the following:

- *Create*: Creating associated objects when an object in a configuration is created.
- *Delete*: Deleting associated objects when an object in a configuration is deleted.
- *Overwrite*: Replacing a configuration in the database with a configuration that has the same identifier, but a new time stamp.
- *Inquire*: Inquiring the associated object about the associated object.

With concepts discussed above, the basic project information pieces are defined by creating libraries of project feature objects. Five types of feature objects are organized into a project model: product feature object, activity feature object, resource feature object, method feature object and participant feature object. The product feature object defines attributes and relationship related to building a product. The activity feature object defines attributes and relationship related to construction activity. The method feature object defines attributes and knowledge of construction method used for the construction of building projects. Resource feature object defines attributes for materials, equipment and labor resource used in the construction process. Participant feature objects define attributes for a project participant.

Project View Object Modeling

An integrated project model attempts to provide a conceptual schema that combines and integrates several functional views related to actors from a variety of disciplines and pertaining to all stages of construction. Each of these views has view specific abstraction need that should be supported in the common model. Although most modeling approaches take abstraction mechanism as their basic approach and use specialization as a general data abstraction, the way these mechanisms are applied may be specific to each view. In general, for each project, there is one facility and the feature object representing the facility has only one physical value but many functional values, depending on the various views of the information by the participants of different discipline. The view object provides the mechanism for aggregating facility components corresponding to a particular discipline. The definition of a view object must be broad enough to enable coverage of all the specific information of a specific construction information view.

A feature object has the great advantage of allowing the formal definition of semantic views and the characterization of specific domains. Furthermore, it permits the formulation of a specific purpose in a systematic and structured manner. A project view object is based on a formal representation of the domain of interest and partially instantiated specifications of feature objects that can be satisfied by more than one product. The project view object is the set of feature object occurrences that interest one of the experts that participate in the construction project.

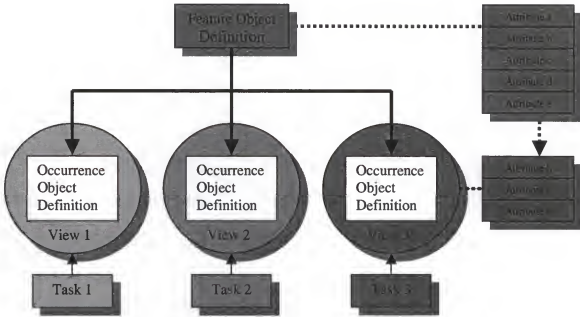


Figure 3-6: Occurrence of feature objects according to view

Occurrence is the second representation of a feature object in the view object, and is derived from feature objects' primary representation. Occurrences refer to the definitions of their primary representation. In contrast to the primary representations, the occurrences of a feature are view-specific and may not be used for reasoning about the topological relations of components across views. Occurrences filter the information based on the special view's stated goal, as illustrated in Figure 3-6. The definition of object occurrences and their attribute groups depend entirely on the needs of the particular view in which the occurrences are used. Some of the attributes of the primary representation can be concealed behind existing occurrences. The influence pattern is a shadow network of attributes. It can be seen from this description that only a portion of the attributes describe a feature object is used at any given view, as shown in Figure 3-7.

Each occurrence belongs to one and only one view, which represents the discipline that uses a certain group of attributes. An occurrence represents the special set

of attribute of a feature object that meets the special interest of the viewer. All project information is modeled as feature objects in the case of multiple occurrences, depending on different points of view for construction information management, for example, contract management, scheduling, cost control, daily administration, or change management. A feature object can in turn be used differently to meet different requirements thanks to a common feature definition and occurrence.

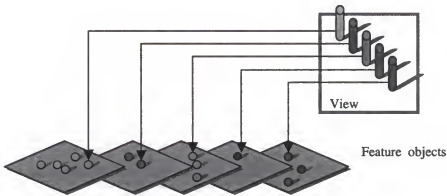


Figure 3-7: The definition of a view object.

Different views may use different shape descriptions, topologies and dimensionalities to express spatial information about objects that are applied in a functional view. Objects may have different characteristics in various views along with different types of information attached to those characteristics. Within one view, one may have different aggregation or abstraction levels at which the same feature resides. The shape representation should be adaptable to these levels of abstraction. Feature object spatial representation may have different characteristics in different views along with different types of attributes attached to those characteristics.

Once the project feature object instance data has been parsed and converted into the domain specific representation, it is stored for use as part of the data of a view object.

Before the data description of the project feature object instance can be stored in the view object, it is first validated to ensure that it is a true subset of the conceptual project feature object instance. The following checks are performed:

- All entity definitions in the view object are matched with definitions in the feature object definition.
- All entity attributes in the view object are matched with definitions in the project feature object definition. Type and name checking are performed.
- The cardinality of each entity in the view object is checked against the feature object definition. The upper and lower bounds in the view object entity definition must lie between the upper and lower bonds of the corresponding feature object.
- The "optional" status of the view object entity is checked.

In order to define the view object in an object library, it is at first necessary to define the purpose of the domain. Then the properties of interest to the viewer may be distinguished, and finally the objects can be sorted into view object classes with regard to the chosen properties. This requires both factual knowledge of the objects of interest and that the purpose of the domain object is carefully considered. Note that every view object has view content which in turn has a data member of all the instances in the domain.

As a view object is application-specific, the design of a view object involves an extensive survey of the sources, users, information flows, and information content of all documents and information used on construction projects. This analysis is summarized as matrices with coordinates that describe construction personnel vs. function, personnel vs. information needs, and information vs. document type. Three views are defined in the project model: cost, process and site view. The process view object represents

construction activities (processes) along with the products they operate on, the participants involved, the sequencing constraints, the resources used, etc. The cost view object depicts a few central entities such as projects, work items, participants, etc. It then identifies a number of the transactions that must be tracked throughout the life of the project. The site view object stores a name, a description, a diagram or picture, and related limitations for conditions. The relationships between products, the processes that can be used to produce them, and the required resources are also stored.

Furthermore, the research challenge of developing view object model that is more complex than the traditional database problem supporting multiple views. Traditional database views derive the data needed for an application from the database, which is only updated centrally. View objects proposed in this research consist of not only the data but also many kinds of methods and functions in order to extract or update the information according to each participant's need. An extension to accommodate this new application requires that every view have a set of methods for specific applications. Using the conceptual modeling approach, domain specific operation can be analyzed through studying the elemental propositions of the problem domain. In this research, most views support applications that generate new data and generate part of the construction process. Various type of procedural operations can also be found in the view object for example:

- Changing the properties and internal states of view objects;
- Analyzing the domain problems;
- Inferring new information;
- Calculating result;
- Maintain dependency relationships; and

- Perform information broadcast and automatic updating.

The overall approach for developing the project model is to represent the construction projects in terms of feature objects and view objects. The project model is defined by creating libraries of object classes. The project object library defines all the classes required to create the objects in the construction project information model. The object classes organized in project objects libraries provide a template for generating the project information database. This step is essential to represent the construction data and knowledge on a form of templates that can be used for organizing project information in an integrated framework. A set of object libraries is developed for the information integration support environment, which defines the representation schemes for information structure and information processing mechanisms.

Since an object oriented GIS has been chosen for the project model implementation, an OO development process is employed for the development of an object library, and is summarized below:

Step 1: Describe and extract requirements and specifications;

Step 2: Analyze and identify the candidate classes/objects;

Step 3: Classify the candidate classes/objects;

Step 4: Identify responsibilities to classes/objects;

Step 5: Identify the variables and methods in each object; and

Step 6: Implement object schema.

The object libraries are established on the data modeling methodology. The project object class libraries define all the classes required to create the objects in the construction project information model. The object classes organized in project objects

libraries provide a template for generating project information database. These classes are used in developing graphical and non-graphical construction information model. In the definition of the project object, object classes are identified and subtypes are declared. Common attributes and behavior of objects are extracted at a higher level for better reusability and interoperability of project objects. After relationships between objects are defined in each object class, the project information objects are integrated to a global framework. The project object libraries are implemented into ODBs (Object Data Base) and implemented in GIS packages. The details about project object library implementation are specified in Appendix A.

System Implementation of Construction GIS

The overall architecture of the proposed GIS-based integrated information system consists of two supporting paradigms: the formal representation paradigm and the implementation paradigm. The formal representation paradigm consists of an object-based, implementation-independent information model for representing the organization of components and activity that constitutes the project together with associated attributes, as discussed in the previous section. A computer implementation of this information model, interfaces with new computer programs described herein to establish the implementation paradigm. The system implementation plan discussed in this section addresses the research objective in developing an integrated information system.

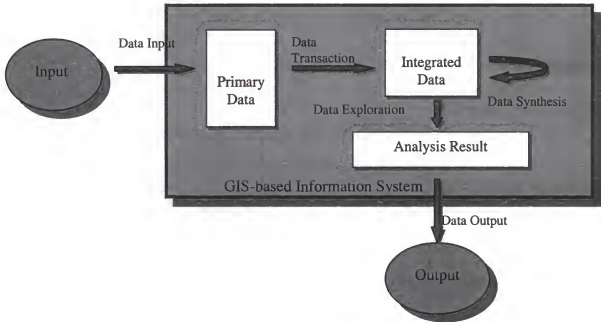


Figure 3-8: A process view of GIS based information integration system.

The theoretical architecture of the system implementation resulted from detailed analysis of information integration process. The process of information integration may be modeled as a series of transformations between the real world, raw data, and the integrated data. Five major processes are identified in the diagram of Figure 3-8.

The process begins with the collection of all basic information or data that are relevant to a certain project. During the information initiation stage, the data for one specific project aspect is introduced, which includes the physical and functional representations of the construction product or/and activities for this phase. These are then input to the integrated system to provide the basis for its digital representation of the real world. Data used in the construction GIS consist of architectural design plans, the actual construction knowledge data, photographs, drawings and notes from job site. The data include actual field data, generated data, or historical data. The data format can either be

text description, numerical value, date/time value, or counter value. One of the main points here is the extent to which data are collected or assembled from different resources in different formats. During the information transaction stage, data are transformed to an open project model via information transformation process. Within each transaction process, a model partition is decomposed to a specific degree of granularity, then synthesized with other model partitions. During an information exploration stage, any piece of information about the project is selected by constraint propagation and satisfaction. Within the system, a vast range of manipulation operations are available to transform further the data to store the results. These may be communicated as tabular reports or graphic displays via hardcopy or screen.

From the process analysis specified above, the overall framework of the GIS-based integrated system requires three main types of activities:

- Collect and model the logical configuration of tasks, resources, and events for a project operation, process or task.
- Perform information synthesis and integration to determine both the potential and actual interaction of individual information items with the modeled logical configuration.
- Sharing the result of integrated information, aggregated according to retrieval criteria. In practice, this produces aggregated preferences of decision for individual professionals.

The overall process is divided into distinct modules, where a module is defined by the type of problem it addresses and the type of solution it generates. Each module is a logical piece of the system and encapsulates the detail of each piece in a set of interface

functions or subroutines. Each module has clearly defined boundaries of responsibility in /the overall system, and clearly defined actions that it can perform. This modularized approach enables the creation of a rich environment that is capable of serving existing and future applications.

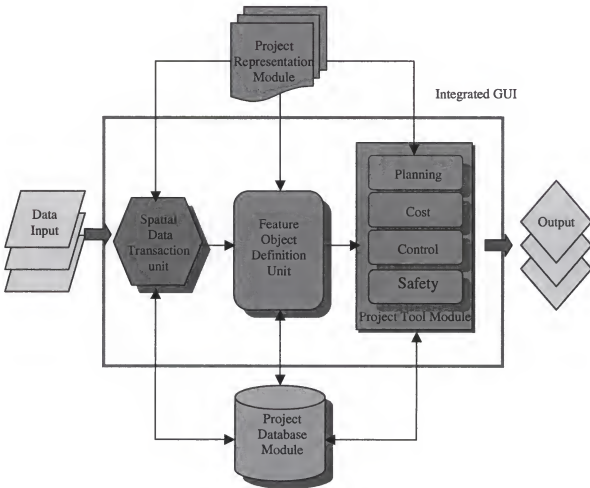


Figure 3-9. Prototype system architecture.

In light of these strategies, the GIS-based integrated information system is developed, schematically shown in Figure 3-9. In the targeted prototype, the following components are distinguished:

- Project central database module;
- Integrated project representation module;
- Project application tools module; and
- Integrated graphic user interface.

These modules are discussed in greater detail as follows.

Project Database Module

The project database is created and initialized in accordance with the project model schema. Four categories of data occurs: a) static or permanent such as CAD and the schedule data, b) process plans generated by the process planning system, c) critical real-time data about the current processes as raw data for the planning & control system, d) resource status for items either in-hand at the site or on-order from different suppliers. For dynamic entities (e.g. concrete delivered to the site or dynamic process activities), a database feature object is created. Several versions of these objects will be generated during run-time to represent changes in parameter values.

A project database is created in three stages. Firstly, it is initialized to the project product and process representation structure described in previous sections. Secondly, it is instantiated by the input through data transaction module. Lastly, it is configured through the representation module to support a specific project management tool. The process of putting data into a project database is referred to as "slotting", since it consists of putting data into a designated attribute field in the database table. The process of exporting data from project database to a project management tool is referred to as

“stripping”, since it maps from a richer global of a project to a more limited local view, where some entity types and relationships need to be eliminated, as illustrated in Figure 3-10.

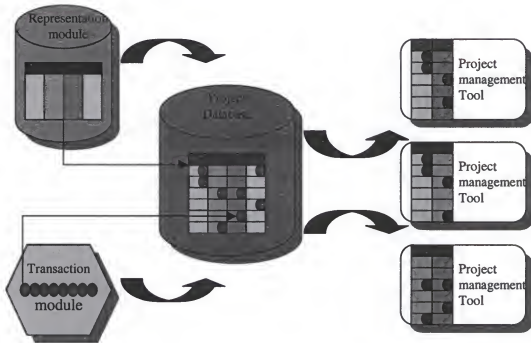


Figure 3-10: The process of project database

The database module proposed for this research is an active, self-actuating object-oriented geo-relational database. The generic ArcView GIS database has been modified for the needs of the database structure implementation. The functional implementation of project object database table is written in Avenue, ArcView’s native development language. Project feature object database tables are defined through Avenue scripts as packages that are comprised of tightly-coupled data and functions that operates on such data.

Integrated Project Representation Module

The integrated project representation module is the hub of the system, whose role is to implement the project models and provide a set of generic data storage and management services for the interaction domain. This module supports the capture and complete representation of the project feature object models. This module also supports the capture of view object models for functional professionals, and supports other necessary modules by providing a unified information structure with which all modules interact. It caters for different data representations, which are encapsulated in feature objects and view objects, as required by the various modules.

The integrated project representation module also supports the collaboration, coordination and communication activities performed within the system. The representation module is in charge of maintaining coherence between the different information in the database and the schema of the integrated data model. It will enable version management of the evolving project models, configuration management of the project model and sub-model decompositions, and dependency management of the relationships between each partial model in the project. For example, if there is any change in project information, the module must decide which set of information should be notified of the change.

The integrated project representation module meets the following requirements:

- Implement the set of conceptual models (object library) that comprise the project model produced.

- Populate the implemented integrated product and process representation with data produced and transmitted to the project model in the form of a feature object library.
- Provide an interface for populating the database (or accessing existing component databases), entering new data, editing existing data, deleting old data and checking the database for completeness and consistency.
- Support shallow control of the information consistency.
- Perform constraint checking at data transaction boundaries.
- Produce view objects from the populated schema in accordance with the individual domain schema.

An implementation of the project representation model potentially consists of a library of functions that are callable from interface program. These functions communicate information between the interface program and the database by means of high-level procedures developed for each of these modules. A complete implementation encapsulates the embedded procedures and therefore offers a uniform interface that is consistent with the proposed object-centered procedural. Two types of procedure that use the object-centered interface library are envisioned. One is an interactive, interpretive procedure for data description and access in an interactive environment. The other supports users as they interactively define and query project database tables. This procedural approach not only provides high-level data abstraction but also hides unnecessary details about representing, defining, and accessing project information.

The integrated project representation module contains three main units: CAD data transaction unit, feature object definition unit and view object definition unit.

CAD data transaction unit

The CAD data transaction unit provides a set of prototypical mappings implemented for conversion and well-defined meaning of CAD abstractions. A spatial definition outlines the set of rules that govern the creation of spatial data from CAD drawing. Any set of standards and procedures used to create these drawings can be encapsulated into this definition. Each of these drawing definitions represents a set of procedures and standards that will contain the CAD data to support creation of spatial feature representation. Although each type of spatial data (2D, 3D, GPS, or just groups of coordinate pairs) contains unique requirements for performing extraction procedures, the underlying methodology within the system remains consistent throughout the platform. Specifically, the extraction process comprises the following tasks:

- Transform the basic spatial data into a GIS coverage;
- Identify logical geometric groupings of objects and extract geometric information for each element within the grouping;
- Define spatial data and transform each element into a feature object representation.

In the system, the process of converting low-level geometric entities in a CAD drawing into higher level objects in a GIS theme, then passing them to the independent data model, is performed in three steps as follows:

- Create a spatial object from a CAD drawing, selecting an item from the CAD drawing. Independent spatial data could be captured automatically from CAD. The

class definition, reference and additional attributes are entered manually via a user interface.

- According to the object definition, the shape of a low-level spatial entity is transformed to the GIS theme. A standard graphical file, in this case a shapefile, is created for each spatial object class. According to the spatial object class definition, the shape type described in the shapefile is predefined. The shapefile is augmented every time there is a new object created.

- Finally the database table of the spatial object is populated. Once the spatial object is created, its low level geometric entities and other relevant information, such as their location, dimension and topological relationship with other elements, are captured and stored in the database table. The extracted object is stored in the class database with a reference that is allocated by user. This reference is uniquely identified with each object and is used by other feature objects to main the integrity of each object.

Once the low level spatial primitives of a project are incorporated in high level objects, the process of populating the feature objects with data begins. Although the spatial object is input only once, it may be represented in many different contexts. Other view objects also use this data to generate new spatial data that is required for specific views.

Feature object definition unit

The purpose of this unit is to provide persistent storage of project representations in accordance with the project feature object model schema. This inputs data for the

feature objects defined in project models which form basis for representation of project information. The object definition process requires both the flexibility to obtain data from a diverse set of origins and the intelligence to transform the data into attribute values within an integral representation. These transformations are the inevitable results of data integration algorithm.

The overall approach is to represent a construction project in terms of feature objects. The project model is defined by creating libraries of object classes. For each class, the attributes of the objects and the relationship that this class of objects can have with other objects are defined. The feature object definition unit uses this information to instantiate the schema data of the project model as discussed in the previous chapter. These instances will be defined in accordance with a schema that is the same as, or a subset of, the schema used to initialize the data store. Each instance must be validated to ensure that it complies with the constraints defined in its object model. When all instances have been added, the resulting model should be validated to ensure that it complies with the overall initialization schema. The process is based on an integration of three problem-solving approaches: the process-based hierarchical decomposition of the information set, the product-oriented automating coordination process, and constraint propagation approaches.

The feature object instance is created by inserting parameters to a predefined object library database. The object has attributes such as class, object id, and property fields associated with the feature object. Since a feature object can be inserted more than once in a particular project, that is, multiple versions of feature object exist in the database, an attribute called status is created to make the most recent version of a feature

object distinct. Thus, a date attribute is created to mark the time stamp for every version. The value of feature object attributes may be instantiated automatically by the software applications or are keyed in manually through a user interface.

A common interface is created for each category of feature object. The common interface is compliant with the object definition and associated database table structure. From this interface, users can set up new feature objects or review information about existing objects, editing changes if necessary. This then starts a transaction on the database, creates an instance of the element in the database, then commits transaction to the database. The advantage of this is that the interface and the database objects themselves can contain knowledge about how the feature object is designed.

View object definition unit

The main objective of the view object definition unit is to dynamically create a view object that reflects a particular construction application view of the data, then integrate this view with construction applications. The project view definition unit, based on the layered information concept, provides a model that can integrate the needed task-based system across different operating professionals and present information, while having a bi-directional link to the database module. This allows to make local use of various items of existing and possibly heterogeneous information and to distribute the project model among several teams, over time. The unit is responsible for migrating project data into the view model environment and retrieving data to feed the project view window. The view object definition unit offers capabilities that range from stepwise construction under the user's control to the automatic generation of new data. These

functions enable an interactive refinement of the project database. This unit also cooperates with the application tools used for the professional analyses performed within the system. In order to participate actively and effectively in each of these modes, view object definition unit therefore contains also a generation and evaluation component that allows users to specify and modify dynamically the project data by:

- Providing library code for accessing the database and requesting input data
- Providing geometric and non-geographic data browser functions
- Providing library functions for aiding in the creation of an editor for

changing and adding to the data

- Providing functions for creating a formatted result file
- Providing library code or an application for informing the user of the

status of the database (e.g., related data change).

The generating of a view object is automatically performed by the system. The view object definition unit interacts with feature object library and database to obtain all the data needed to generate the view object corresponding to the application's view of the data. The system also maintains the integrity of the data by maintaining the relationships that exist between objects in the project model. The unit is supported by the project object library, enables intelligence to be incorporate in the system, makes it possible to obtain any information about any object, obtains the most recent data and builds the view object dynamically. In order to have the latest information available for preparing view objects, well-planned data collection and good modes of communication are required. Also it is necessary to build the most updated view object. The view object result is changed over

time and only the most updated version of feature objects are used for building the view objects.

Three major views are created by this unit. The process view unit automatically produces objects that represent a generic set of construction activities that can be allocated to a specific product. Data about resources, construction methods, productivity rate, duration, etc. are allocated to process view objects. Dependencies between activities are also determined depending on the precedence relationship. The resulting network of activities with their technological and organizational dependencies is generated automatically. The spatial representations of the project feature objects are used and transformed to the view object according to the view definition. A suitable graphical environment is included for data mining in support of this process.

The site view unit dynamically produces site view objects that carry project performance information. Information regarding duration, progress of the activity, material, equipment, and participants involved in the product building activity, etc. are extracted from project feature objects. The functional systems and the tasks required for their installation can be determined automatically by the intelligent system with or without interaction with the user. Such information is displayed and manipulated in graphic form, where the user can interrogate the graphical objects to obtain more information.

The cost view unit generate cost view object via its integration with project feature objects. Information from feature objects such as product elements, their dimensions, associated activity, resources used in construction project, etc. are collected and the construction cost are calculated from such information. The allocation of

resources to activities and their final scheduling can be performed automatically per analytical objectives. A graphical representation of the resource use is developed which enables user to explore a realistic graphical representation simply by interacting with the graphical representation.

Project Management Tool Module

Implementing useful construction applications is part of the purpose of the research on GIS-based integrated information system. Through these applications the ability of our approach has been proved in practice. An added benefit of these tools is demonstrating the capability of integrated information environment and future direction. To date, prototype system development has not substantially addressed application tools development. However, we have defined the target domain to prove the research concept with conventional applications and carry out several prototype applications, these applications have been selected to reflect their potential role within a larger context of construction management application. The application areas addressed by these systems are as follows:

- Construction planning and process analysis;
- Project estimate and cost analysis;
- Project control and interim evaluation.
- Project safety analysis and evaluation.

To perform like an expert analyst, project management tool module is structured according to three types of knowledge: a problem is described using environmental knowledge; procedural knowledge is used to help design a solution process and

determine the values of parameters in models; and structure knowledge (the steps of algorithms and the data structures on which they operate) is used to solve problems.

Construction planning and process analysis tool

The main objective of construction planning and process analysis tool is to dynamically analyze construction activities, generate construction plans of a project based on the required activity and the availability of resources, and generate an as-built schedule based on the site condition. Additionally, the user could analyze the construction project progress as derived by the construction plan with the aim of identifying activity delays. A set of analysis functions built into the tool include:

- Identifying work items,
- Calculating take-off of work package,
- Calculating work duration,
- Analyzing the precedence of activities,
- Sequencing activities,
- Generating construction plan,
- Analyzing timing differences between an as-built verses an as-planned schedule, and
- Generating construction activities progress report.

Construction cost analysis tool

The main objective of cost analysis tool is to assist preliminary cost analysis and produce cost estimate according to different estimate classes for package quantification

and cost aggregation. The process of sequentially updating the parametric information to obtain progressively more accurate estimates of a construction constitutes the basis of the sequential parametric estimating (Omimi, 1995). At any given stage in the development of a project, parameter values defining one or more work-packages can be drawn from project feature objects and cost view object. Cost estimates of the work-packages that can be defined by known parameter values at any given stage are summed up to determine the parametric cost-estimate of the project at that stage. Cost estimates are sequentially updated as more parametric information become available. A set of analysis functions included in this tool are:

- Identifying cost items,
- Calculating quantity needed for cost items;
- Identifying cost item unit price,
- Calculating costs for each cost items,
- Summarizing estimates, and
- Analyzing cost at different levels of abstraction,

Construction project control tool

This project control tool supports project on-site condition monitoring. The project monitoring visualizes the logical configuration of tasks, resources, and events for a project operation, process or task both for the initial plan or the actual results; perform sensitivity analyses to determine both the potential or actual effects of changes in the logical configuration of the product, process and resource allocation. A set of analysis functions in this tool include:

- Identifying site items,
- Reporting site item information,
- Checking image and other multimedia file associated with specific site item,
- Checking specification associated with specific item,
- Performing change and change sensitive analysis, and
- Generating performance reports.

Construction safety analysis tool

Safety analysis tool can be used to provide the accident prevention information according to project condition to assist the on-site safety management during the execution of a project activity. Each construction work has similar contents of work or situations where similar accidents have occurred. This information about past injuries might be useful to make predictions about the number and types of injuries that are likely to occur in the future. (Hinze and Russell, 1995). Accordingly the safety analysis is based on projecting hazardous work conditions using historical accident information. The safety analysis deals with the handling the project information, historical accident information, and principles acquired through experience and association. The safety analysis tool can be used to evaluate the site conditions and to detect the potential accident conditions. A set of analysis tools are built in this tool include:

- Retrieving past accident data,
- Evaluation project condition,
- Searching for common keys in historical accident database,

- Projecting hazardous condition in the project,
- Indicating the potential accident location, and
- Reporting the accident projection and prevention measure.

These tools offer several advantages. Firstly, the application is isolated from the project model and database, which allows considerable flexibility when enhancing both the database and the application. Secondly, the application can manipulate data using the project data without knowledge of how the data is stored in the database. Finally, an application that is built with a tool interface can be easily switched to work. This module is intended to include more application tools with future development.

Integrated System Interface Module

The role integrated system interface module provides management of user interfaces. The integrated system interface module supports the development of consistent and homogeneous interfaces for both global and local user interfaces as well as for the individual modules of the system. The objective is to allow interfaces to be built on top of other modules that allow the user to view the project model and to select graphically only those parts on which they wish to work. It is designed to have multiple windows that inter-communicate with each other and compose a tightly integrated user interface. The emphasis for the entire user interface design is on simplicity.

There are six windows in the system: (1) CAD object definition window, (2) feature object definition window, (3) site view window, (4) process view window, (5) cost view window and (6) portfolio window. Each window appears to the user as part of a

unified whole. A common interface is developed for the windows based on uniformed user interfaces. The common interface includes command menus and data displays.

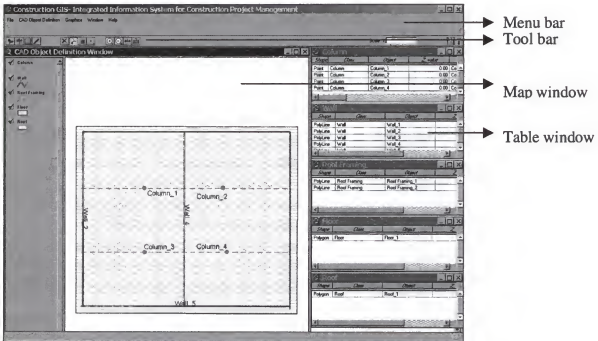


Figure 3-12: An example of the window user interface

Figure 3-12 shows an example of the window. As with most Microsoft Windows programs, the top part of the window shows the name of the modeling package and, below that, the main menu. Below the menu, is a toolbar consisting of buttons that can be pressed to perform actions. Below the toolbar is the main display, showing the map and tables.

A command driven model of interaction is chosen for the interface. This is a standard way of implementing a GIS application, with extra sets of menus being added to the standard software hierarchy to invoke new commands. As various commands are

called to add special functions with each system modules, each command invokes a function in the application which starts a transaction on the database, prompts the user for parameters, positions and displays the elements, creates an instance of an object in the database, or calculates data and performs a user request, etc. The display is then updated to reflect the result of the command function. The command menus also contain commands for changing the window, zooming, performing graphics editing, etc. The commands are carried out in menu items such as menu bar, buttons in a button bar, and tools in a toolbar.

The display normally includes two main spaces: table space and map space. To manage complex information to user in an easy-to-understand way in a limited display area, the interface design efficiently assimilates all the available information through a condensed format. Table space depicts the database parameters of a specific object while map space is a cartographic representation of the output of the project model. The two spaces are inter-linked and the user is able to view these spaces simultaneously. The graphic objects in map space and database objects in tabular space are associated. The map space is a graphical viewer, which provides the simplest way to access data for browsing, querying and updating of data, choice of subsets out of the project model. The map space provides a number of capabilities. The first capability is the generation of high-resolution cartographic displays. These displays are also supplemented with other forms of graphics, including symbols, which will be useful for exploratory data analysis. More specialized graphics will be necessary for depicting the results from analytical models and sophisticated statistical techniques. Tabular space supports graphical capabilities of map space. Reporting is a core function of the table space. Besides the

major two display spaces, there are also charts, image windows, identification windows, and text windows that can display data exploration and analysis results.

The advanced user interface design takes advantage of multi-tasking and built-in coding. The map and table are intelligent, i.e., control the displayed data it displays and associated menu commands. For example, when the user starts an application window, the database is opened and the available relevant data is loaded with the map and tables. The command menus are automatically updated to support specific application. When the user leaves the application, the database is closed and the current display is discarded. The next time the application is started it will recreate the display with the most up-to-date data from the object database. Details of exactly how the interface carries out this task are included in Appendix B.

Based on the system architecture, the aforementioned modules are built into GIS system architecture to perform integrated information system functions. The selected GIS system is the Environmental System Research Institute (ESRI) software ArcView Geographic Information System. ArcView was selected among the other possible GIS systems because it has a powerful development language called Avenue. The project database is implemented as tables in combination with spatial themes. The project model is implemented in Avenue scripts. The integrated system facility is developed based on ArcView graphic user interface consisting of individual but inter-communicating units supporting various functions and activities of the integrated system.

System Case Testing

The prototype is considered as a developmental model of the system for testing purposes. After installation of the coding into ArcView, the prototype system was tested for reliability and quality. The objective of testing is multi-dimensional. The main focus is on those approaches for project information integration that optimize the project information analysis and management. Time and resources do not permit, nor is it appropriate, to test every aspect of the system. What are tested are those functions that are directly related to the information integration process and how the system is used. Important aspects of the testing include the following: the effectiveness of the project modeling approach, the effectiveness of system development and feedback on future development of the system. The proceeding criterion relates to how the prototype system and proposed research approach meet the practical needs of construction project information integration, and therefore the usefulness of a GIS-based integrated information system such as Construction GIS. The rough outline of the test is:

- Building CAD spatial objects,
- Building project feature objects,
- Performing analysis of construction planning,
- Performing analysis of project progress,
- Performing preliminary cost analysis,
- Performing on-site condition monitoring,
- Performing preliminary analysis of change order, and
- Performing preliminary analysis of safety on site.

All these activities are aided by the Construction GIS system interface windows:

- Spatial objects are built with the system CAD Object Definition Window.
- Feature objects are built with the system Feature Object Definition

Window.

- Analysis of construction planning is conducted through the system Process

View Window.

- Analysis of project progress is conducted through the system Process

View Window.

- Preliminary cost analysis is conducted through the system Cost View

Window.

- Project change order analysis is conducted through the system Site View

Window.

- Safety analysis is conducted through the system Site View Window.

- On-site condition monitoring is conducted through system Site View

Window.

Details of user interface functions are included in Appendix B.

Building Spatial Objects

The test starts with building spatial objects, using the CAD object definition window. Through this window, the user selects an object from the CAD drawing and a definition dialog prompts the user to enter the spatial object definition, as shown in Figure 3-13. From the dialog, the user could select the object class and give a unique identification number for each object. After the definition is finished, the spatial

object is established and the spatial database is set up. The spatial objects are built and the database tables are set up. Through the CAD object definition window, a user could also edit the spatial object definition, edit the shape of spatial objects and add new spatial objects that are not included in the CAD drawings. When adding a new spatial object that is not included in the CAD drawing, the user first selects an object class, which defines the output spatial object shape. For example, a column will be referred to a point spatial shapefile and a wall will be referred to a polyline shapefile. Also the user needs to input the object definition along with the spatial shape at each new object input.

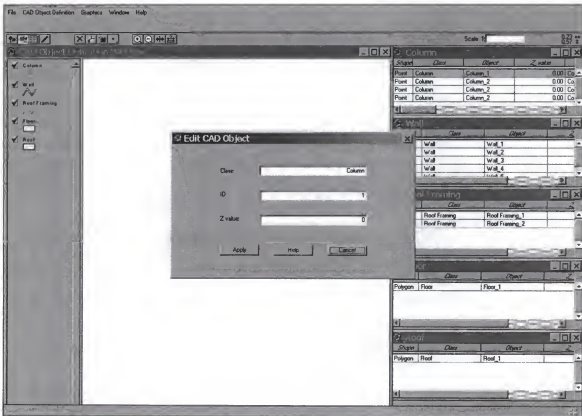


Figure 3-13. CAD object definition window interface.

Building Feature objects

The feature objects are built with feature object definition tools. User entry through the interface is input into the database and the project database is populated. The product feature objects are built through the product feature object definition interface, which is shown in Figure 3-14. Through this interface, (1) user selects the class from a list of predefined classes; (2) inputs object id number; (3) selects the type; (4) selects the material from a predefined material listing pool; (5) selects measurements; (6) inputs the picture items if there are any, and (7) builds the *compose_of*, *connect_to* and *conflict_with* relationship if necessary. The data is then saved as a new record to the database, since the same product feature object could have multiple versions.

The resultant product feature object table also has version status and a database time stamp for each version of the object. In order to separate the most up-to-date version from the old versions, object status codes are assigned to each version. Only the most updated version of a feature object has the version status indicating this record is the most updated record, whereas all the old versions have the version status indicating this record is not the most current one. When a new record of object is entered into the database table, the version status is set to most update, and the old record version status is automatically changed.

Also through the product feature object definition dialog, the product feature object could be edited. When the user selects the class and inputs the object identification number, the system will search through the existing database and retrieve all existing attribute values of the most recent version of the feature object. These values are then

input to the user interface by which the user can edit the necessary values and save new values to database. In this case, the version status remains unmodified, but the database time stamp will be updated to the new time.

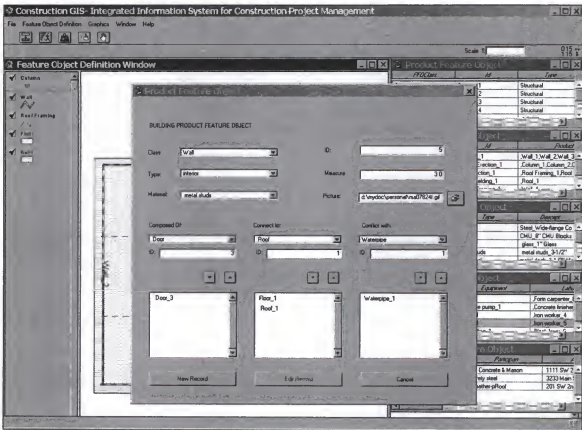
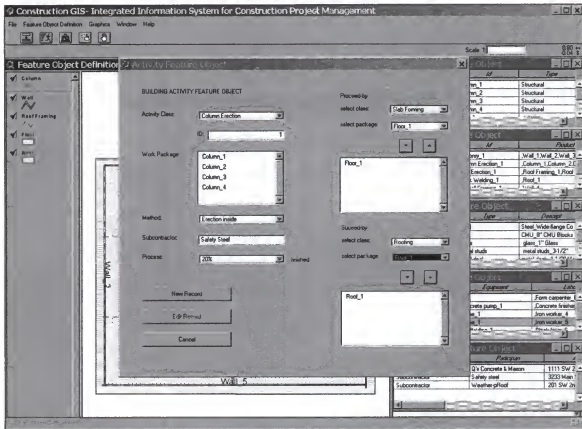


Figure 3-14. Building product feature object interface.

The activity feature objects are built through the activity feature object definition interface, which is shown in Figure 3-15. Through this interface, the user (1) selects the class from a list of predefined classes; (2) inputs object id number; (3) selects work packages; (4) inputs subcontractor; (5) inputs the work completion percentage; and (6) builds the dependency relationship if necessary. The work package is selected from a list,

which is derived from product feature object tables. Also, the activity feature object definition dialog can be used to edit existing database records.



database. Similar to the product feature object table, multiple versions of activity feature objects are kept in the database table with associated version status and time stamp.

The resource feature objects are built through the resource feature object definition interface, which is shown in Figure 3-16. By a process similar to that is used to build product feature objects, the user (1) selects the class from a list of predefined classes; (2) selects resource type; (3) inputs provider; (4) inputs unit of cost; and (5) inputs the quantity purchased (different from quantity needed). The attribute values are then saved to the resource feature object database. Also, the resource feature object definition dialog could be used to edit existing database records.

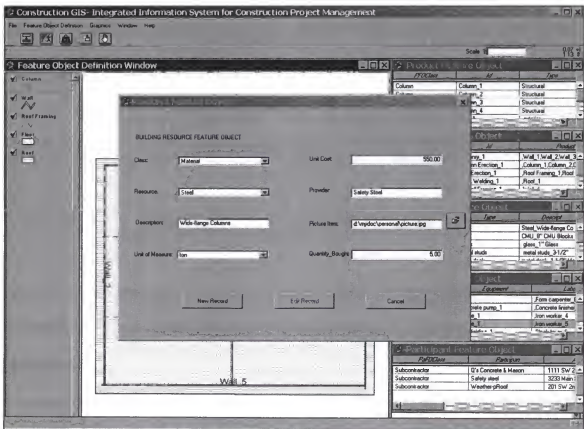


Figure 3-16. Building resource feature object interface.

The method objects are built with a method feature object definition interface, as shown in Figure 3-17. Through this interface, a user first selects the activity. A list of possible methods is displayed in the method selection combo box. From this list, the user selects the specific method to be defined. For each method, the user enters equipment used in the method, quantity of the equipment, the labor used in the method, quantity of labor, and the productivity rate with the selected equipment and amount of labor. The attribute values defined by the user are then saved to the method feature object database table. Also, the method feature object definition dialog could be used to edit the existing database record.

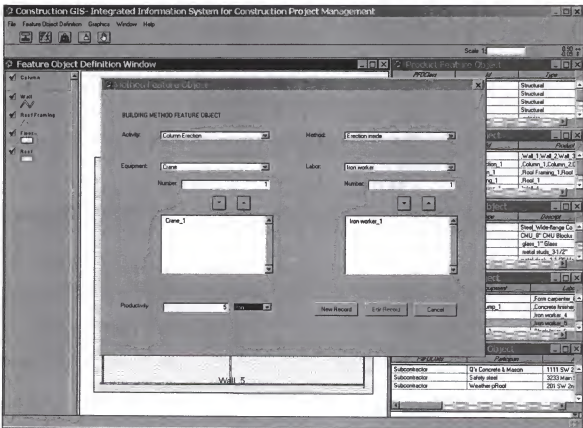


Figure 3-17. Build method feature object interface

The participant feature object is built through the participant feature object definition interface as shown in Figure 3-18. The user first selects the participant class as subcontractor, material provider, or equipment vendor. When selecting the subcontractor, the system searches through the activity feature object table and retrieves a list of subcontractors by the subcontractor field in the activity feature object table. Then the user can select a specific subcontractor and input the contact information and contract information.

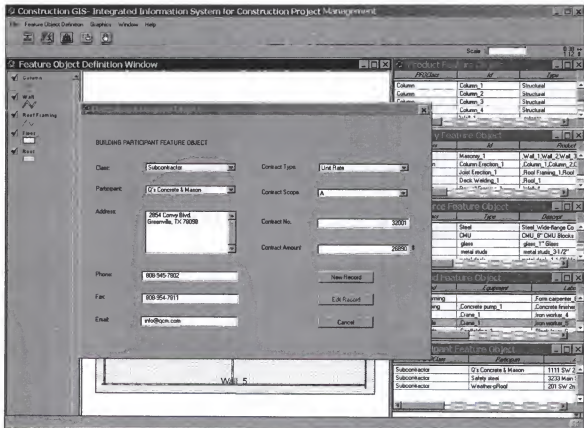


Figure 3-19. Building participant feature object interface.

All the resultant feature objects tables are shown in the feature object definition window along with the spatial objects defined. The database table and the spatial object themes are inter-related. When defining a new feature object, the spatial object theme could be used as a road map. The window is also dynamic: when adding a new feature object, the map display will be updated with the new record added to the display; when identifying on each record from the database table, the related spatial object on map display will also be highlighted.

Analysis of project planning

Once a project model is created by the feature object definition module, a series of construction tasks associated with products are instantiated in the object oriented database as there is an process view object that supports the planning application. Given the technological dependencies between activity objects, their location, and the hierarchy of planning premises, it is possible to derive algorithmically their network, allocation of resources, and schedule. Project planning analysis takes advantage of project feature object database and the library coding of process view object definition. Automated construction planning is based on the definition of process view object and includes the following steps:

1. Generation of work items. Tasks are allocated by the system from the project feature object database to each work type. The related records in the product feature object table, activity feature object table and method feature object table are used to generate work items on the basis of work assigned for the installation of products using

pre-specified methods. The library of process view objects defines how to map feature objects into specific work items. The resultant work items are built as spatial themes loaded into the process view window. Each resultant work item theme will have the attributes of work item, product element, work quantity, equipment, labor, material, productivity rate, work progress and precedence relationships.

2. Determination of work duration. For each work item record, with the productivity rate and workload quantity, the duration of each work item can be calculated.

3. Determination of work sequence and allocation of timing. Work sequence is determined by the system based on the precedence relationship of the various activities for all the work items. With an estimated duration value, timing could be calculated for each work item record in terms of start time and end time as a plan. According to the plan, a schedule bar chart is created. The construction planning analysis result are illustrated in Table 3-2 and Figure 3-20.

Table 3-2 Example of construction planning analysis result

WORK	PROD	DURATION_P	START	END
Slab forming planned	Floor	1	0	1
Slab pouring planned	Floor	3	2	5
Masonry planned	Exterior Wall	12	8	20
Column Erection planned	Columns	2	6	8
Joist Erection planned	Roof Framing	3	20	23
Deck Welding planned	Roof Deck	3	23	26
Install roof Insulation	Roof	5	36	41
Install roof Cap Sheets	Roof	10	26	36
Drywall Framing planned	Interior Wall	6	36	42
Hanging Gypsum Board pl	Interior Wall	4	42	46
Glass Framing planned	Storefront	4	20	24
Glazing planned	Storefront	4	36	40
Plastering planned	Exterior Wall	12	42	54
Total Project planned	Total	54	0	54

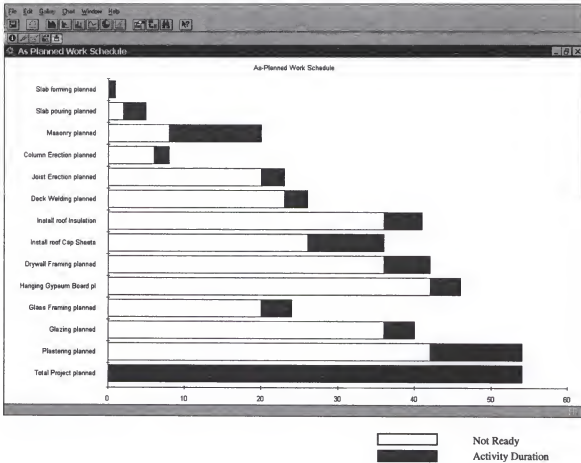


Figure 3-20. As-planned schedule bar chart.

Analysis of project progress

Analysis of project progress is based on comparison of as-built timing and planned timing for construction activities. Based on the user input for activity feature objects over time, the activity start-time and end-time are collected. The activity progress is then compared with the planned timing, as follows:

1. Derive the as-built start time and end time progress reports for each individual activity. Among all versions of the activity feature object, the progress value is

changed over time as the user updates the activity feature object. The progress value combined with the database time stamp (for object version records) is used to derive the as-built timing information.

2. Calculate the as-built duration for each activity using the as-built start time and end time.
3. Compare the as-built end time and as-planned schedule. The result of the project progress analysis is output by the system as a table and a bar chart with as-planned and on-site timing information, as illustrated in Figure 3-21.

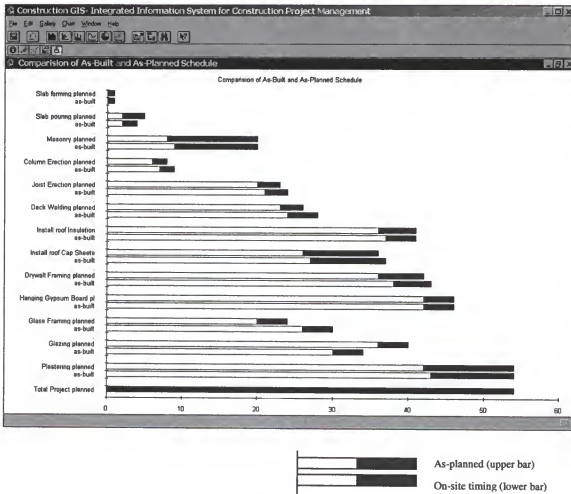
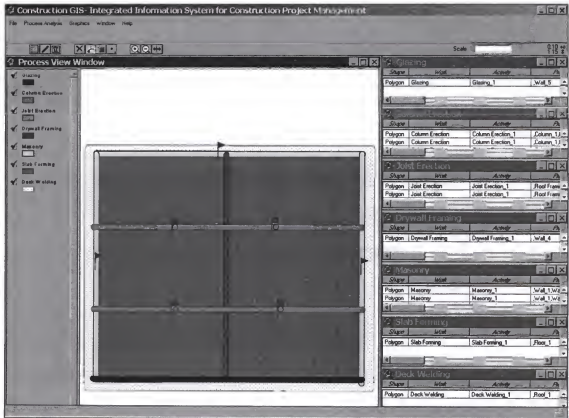


Figure 3-21. Comparison of planned schedule and on site work timing.

4. Another function of project progress analysis is visualizing work progress.

The progress of work activities can be calculated according to site condition and inter-relationships of activities, then visually displayed with different symbols for different work progress, as shown in Figure 3-22. The different symbols on the map indicate activity progress as one of finished, ready-to-finish, in-construction, ready-to-start and not-start. This can be used as a reference for revising, closing or opening works.



- Work ready to finish
- ▶ Work ready to start

Figure 3-22. On site work progress.

Preliminary cost analysis

Once feature objects are created, cost view object will read feature object database and create instances of work items in the database and populate the attributes of quantities and rates. Quantities are derived from the product components whereas rates are established from the work items and resources. Cost analysis can then establish the building project's total cost. Any changes to the project are notified to objects within cost view object in order to ensure consistency of information. The user can view the resources associated with work items and change some of the costing parameters such as price, productivity, etc. The preliminary cost analysis is created by the system to convert a feature object database describing the test project into a list of specific cost components, with which unit prices, units of measure and quantity can be associated to form a cost estimate. The rough procedure for preliminary analysis includes:

1. Generation of cost components. The cost components are in three major categories: material, equipment and labor. For each category, the corresponding project feature objects are searched and attributes are retrieved to form the cost component database. The library of cost view object defines how to map feature objects into specific cost components. The resultant cost component has the attributes of cost component type, product to which cost component applies, work on the product element, quantity needed, quantity purchased, unit cost, unit of measure, provider and picture items.

2. Calculating costs. While each cost component record has the unit cost attribute and quantity attributes, the cost for each cost component is calculated. The cost

of the cost item is calculated by using the equation: $\text{cost item cost} = (\text{Quantity}) * (\text{Unit-Cost})$. This process is repeated for all the cost components.

3. Arrange cost components into estimate. The resulted cost estimate is reported in the cost estimate table, as illustrated in Figure 3-23.

Construction GIS - Integrated Information System for Construction Project Management

File Edit Table Field Window Help

1 of 1 selected

Cost Estimate

Description	Category	Unit	Unit Cost	Alt. Unit Cost	Alt. Description	Quantity	Cost
2-8 Form Work	Material	Slab forming	LF	0.85	Q's Concrete & Mason	1400	1120.00
Form Carpenter	Labor	Slab pouring	Man/day	210.00	Q's Concrete & Mason	24	5040.00
300-PSI Road-base	Material	Slab pouring	Cube-Yard	70.00	Other Materials	120	8400.00
Concrete Pump	Equipment	Slab pouring	Day	200.00	Q's Concrete & Mason	0	0.00
Concrete Finisher	Labor	Slab pouring	Man/day	210.00	Q's Concrete & Mason	26	5460.00
8" CMU Blocks	Material	Masonry	Piece	0.85	Other Materials	8000	6800.00
Blocklayer	Labor	Masonry	Man/day	250.00	Q's Concrete & Mason	30	7500.00
Scalloiding	Equipment	Masonry	LL/MP	8000.00	Q's Concrete & Mason	1	8000.00
Wide-Range Columns	Material	Column Erection	ton	600.00	Safety Steel	5	3000.00
	Material	Steel Erection	ton	650.00	Safety Steel	17	11050.00
Bar plate							
1-1/2" Metal Deck	Material	Deck Welding	ton	220.00	Safety Steel	5	1100.00
Iron Worker	Labor	Joint Erection/Deck Welding	Man/day	350.00	Safety Steel	20	7000.00
Welding Machine	Equipment	Joint Erection/Deck Welding	Day	100.00	Safety Steel	5	500.00
T-Box Crane	Equipment	Column Erection/Steel Erection/Deck	Hour	120.00	Construction Crane	20	2400.00
2-5" Insulation Box	Material	Roofing	SF	4.25	Weather-ghost	12000	51000.00
Exhaustion Cap Sheet	Material	Roofing	SF	4.25	Weather-ghost	12000	51000.00
Lull	Equipment	Roofing	Day	80.00	Weather-ghost	15	1200.00
Roofers	Labor	Roofing	Man/day	210.00	Weather-ghost	50	10500.00
5-1/2" Metal Studs	Material	Drywall Framing	SF	2.50	All Drywall	1160	2900.00
DRY Ceiling Board	Material	Drywall Framing	SF	1.50	All Drywall	3600	5400.00
Drywall Framers	Labor	Drywall Framing	Man/day	250.00	All Drywall	24	6000.00
Drywall Finisher	Labor	Drywall Framing	Man/day	250.00	All Drywall	20	5000.00
Exterior Lbr	Equipment	Drywall Framing	Day	300.00	All Drywall	10	3000.00
Plaster Materials	Material	Plastering	SF	2.00	Sunshine Shucco & E	8000	16000.00
Plaster men	Labor	Plastering	Man/day	220.00	Sunshine Shucco & E	72	15840.00
Scalloiding	Equipment	Plastering	LL/MP	8000.00	Sunshine Shucco & E	1	8000.00
Scalloiding Frames	Material	Glass Framing	SF	15.00	New Horizon	1000	15000.00
Glass	Material	Glassing	SF	12.00	New Horizon	1000	12000.00

Equipment	Element	Unit	Labor	Element	Unit	Material	Element	Unit
Concrete pump	Flow_1	Slab Pouring_1	Form carpenter	Flow_1	Slab Pouring_1	metal studs	Wall_4	Drywall Framing_1
Scalloiding	Wall_1	Masonry_1	Concrete finisher	Flow_1	Slab Pouring_1	glass	Wall_2	Plastering_1
Scalloiding	Wall_1	Masonry_1	Block layer	Wall_1	Masonry_1	Bar joint	Roof Framing_1	Joint Erection_1
Scalloiding	Wall_1	Masonry_1	Block layer	Wall_1	Masonry_1	Bar joint	Roof Framing_2	Joint Erection_1
Scalloiding	Wall_1	Masonry_1	Block layer	Wall_1	Masonry_1	Concrete	Flow_1	Slab Pouring_1
Scalloiding	Wall_1	Masonry_1	Block layer	Wall_1	Masonry_1	metal deck	Flow_1	Deck Welding_1

Figure 3-23 Preliminary cost estimate result

Project on-site condition monitoring

The re-structuring of construction products and processes is a prerequisite in project control. The information requirements for project control can be categorized into

general project database and support knowledge. The general project monitoring includes exploration of the following information: the products; the processes or actions; the resources; the on going site condition records, the pictures of site condition and the regulation and specification that must be enforced. Also it is a nice feature for project monitoring to check the history of project by looking back the condition on site. The project monitoring is implemented as following:

1. Construct site condition. After a detailed project model is generated as a result of the project feature object building process, construction site condition is generated automatically. The project site condition is constructed and the site items are built with the information from the project feature object database. It is spatially represented with spatial representation of product feature object, but with more attributes such as product element, work, equipment, labor, material, picture item, work quantity, progress etc.

2. Explore site condition. The resultant site items are used by the user to explore the site condition and project progress. User could use identification tool to click on each individual site item on map display to get the detailed information. If there is a non-empty picture item, the user could use the image link tool to retrieve and display the picture of the site item, as illustrated in Figure 3-24. These tools provide the basic data exploration capability. Advanced data exploration can be done through the database query provided by the interface. The database query tool requires the user to input the query standard into the query box. Then the system will search the database to find the records that meet the standard. The result will be shown with highlighted database table row and highlighted map object.

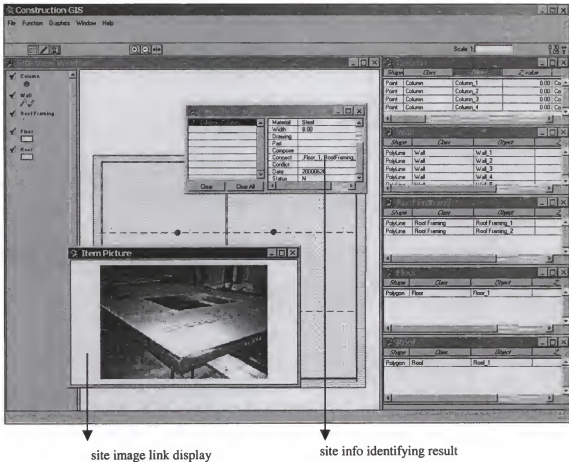


Figure 3-24. Site item identification result and image link.

4. Check product specification. Site view window also has tools for checking product specification. When the user selects the site item on maps, the system will look for special section of product specification related to the selected site item. The process is key word searching. For each product class, special specification key word is pre-defined. This key word is used to search specification document section with content designated for specific product class. These sections then are retrieved and displayed through text window, as illustrated in Figure 3-25.

5. Construct past site condition. The site monitoring is not limited to the present site condition, but applies to the past as well. Past site condition analysis is realized by searching the user input of feature objects over time to get the user-specified time period of the feature objects and get the as-then status of the whole project. Analysis is achieved by user inputs a time interval specification for checking condition. With this time frame, the system searches the project feature object database and gets the version input specific to that time span. The site condition is reconstructed according to information output from the database.

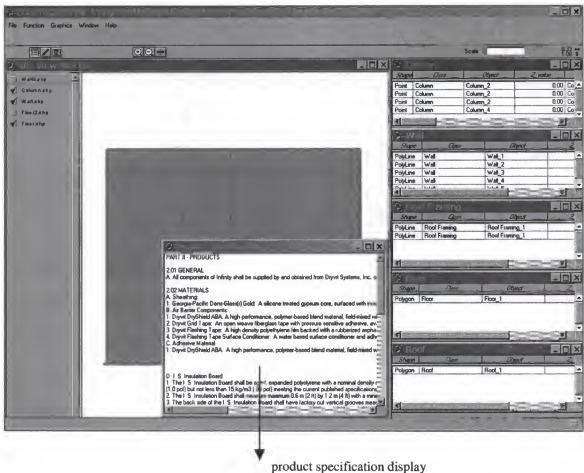


Figure 3-25. Example of product specification checking display

Analysis of change order

As the project goes on, the information about construction activity is updated constantly. These updates rely on the actual progress of the project and the changing field conditions. These conditions may include such elements as change orders, schedule changes, and availability of resources in the field. Changes in the commissioning and the pre-commissioning schedules would also be included in this update. When a specific item needs to be changed, user could manipulate the change directly through the map if the change is spatially or from database entry if the change is on non-spatial attributes. It is done through the following steps:

1. User could edit the spatial location and/or spatial dimension of a product item by editing the product feature object spatial representation. User starts editing mode on the base shapefile of the spatial representation. The change of location and/or dimension is done on the map. When a change needs to be done on non-spatial attributes of any project item, the change is done through database entry. User starts editing mode on the base table and edits table records.

2. Once change is input, change sensitive analysis is done for potential influence. As feature object supports the creation of "intelligent" project databases, since project data are stored as objects combined with rules that respond to events and reacts to context, the change event on individual data is propagated and notified to related data. For example, a "wall" object can inform other objects that its dimensions have changed or can change its dimension if alerted to a change in the connected floor. The consistency and constraint are checked automatically by firing up the update event on specific project items. If there is conflict because of the change, there is a flag indicating that the

transaction could not proceed. If no conflict is found, conversion on related product feature object will be shown on the display and the difference is indicated, as shown in Figure 3-26. After user confirms the change editing, change on the product item is committed. If user chooses discard change, change on project item could be reverted.

3. As cost view object, process view object and site view object are built on the fly based on feature object; any data change on project item will be propagated to changes in resource allocation; and/or changes in the duration (or duration distribution) of specific tasks.

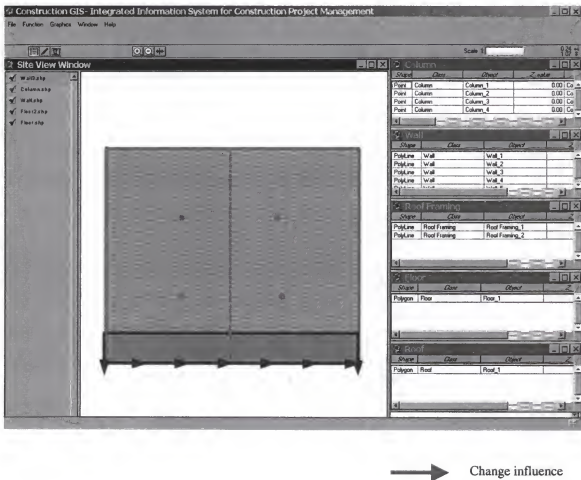


Figure 3-26. Example of change analysis result

Analysis of safety on site

Analysis of safety on site comprises of the acquisition of historical accident data, as well as the use of historical data in inference and analysis. It involves analysis of project information, searching for relevant historical accident key, and projection of the potential accident location and situation. The following steps are used:

1. Retrieve the past accident information. The historic accident database has detailed historical record kept especially for the accident inference purpose, and on direct observations of construction operations, processes and work tasks. The basic structure of historical accident database is hierarchical in order to provide retrieving and updating of the past accident data according to the project progress. The keys for data retrieving include major information for the common condition searching. The basic items in historical accident database include: type of construction, type of work, progress of work, form of work, type of labor, type of equipment, type of accident, cause of accident, place of accident, degree of damage, and precaution measures. The first five items is used as searching keys.

2. Evaluate basic site situation of individual activity. The safety analysis tool retrieves data from a project database using the single key of activity. The basic data items are type of activity, type of construction method, type of labor used, type of equipment used, progress of the activity, place of activity, and spatial condition of activity. These data is used for common key in search historical accident database. Retrieved data is used to project any hazardous area and report preventing measures.

3. The result of the analysis is shown in two ways. First, the potential accident area will be displayed on the map. Plots and conditional displays of results from the inference calculation provide graphical representation allowing for quick and clear recognition of the exception conditions. The hazardous zone and accident-prone activity is displayed on map. Figure 3-27 gives an example of the resulted display of accident-prone places. Secondly, reports are produced on past accident that describes the detail information about the accident and suggestions for reducing the hazards is also given.

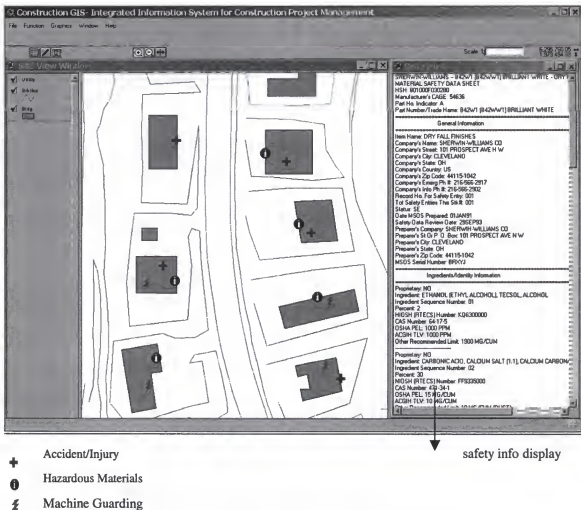


Figure 3- 27. Example of on site safety analysis result

The test procedure has been repeated on several occasions for testing and refining the proposed GIS-based integrated information system methodology. Important lessons are learned and necessary adjustments to the prototypes were identified and implemented. Whereas the present phase of research testing will end with preliminary testing, we consider the testing of prototype system as efficient and serving the research purpose.

CHAPTER 4

DISCUSSION AND SUMMARY

This research demonstrated adequate utilities of GIS technology on getting relevant construction project data in place to effectively support construction project information integration. Methods used in this research also suggested improving the efficiency of the linkage between GIS and models include implementing various database management strategies and interfaces within a GIS environment.

Research Findings

In this section, we discuss our research findings with respect to GIS-based integrated approach, GIS-based integrated project modeling framework and GIS-based integrated system development.

GIS-based Approach

In order to assess the result of this research as a preliminary step towards GIS-based integrated system, it is useful to distinguish how GIS-based integrated system meets the functionality of construction information integration and challenges of designing an integrated information system (per chapter 1). The methodology of using GIS-based integrated information system for information integration has been shown

useful and efficient and meets the research challenge in four areas: information environment, data representation, function set, and system configuration.

First, GIS-based integrated systems meet two mechanisms of information integration: horizontally in that it supports integration of information from different project actors/data, and vertically in that it supports the information integration through project stages. Data environments developed for the GIS-based integrated system for information framework and analysis purpose have been valuable for integrating different data sets through open data structure techniques. A major strength of the GIS-based integrated system is that it can accept and merge diverse data into a single database, giving the user a flexible and powerful set of data from which to work. This is a special strength of GIS that few technologies share. Such a framework can accommodate various types of data and provide multiple accesses, so integration of heterogeneous data is readily achieved.

Secondly, the generic data representation structure in GIS provides a powerful solution to information integration. While emphasizing the role of spatial representation, the binding power of geo-relational model over the spatial data and non-spatial data is another advantage of the data representation in GIS-based integrated approach. The GIS-based integrated solutions extend geometrical description of the construction project with further information within the fields of cost estimation, construction planning, and project management. By tightly combining spatial and non-spatial data, the proposed GIS-based approach provides means of computationally representing, organizing and linking information about constructed facilities. The generic data structure recognized the fundamental and important interrelations between the spatial and non-spatial attributes of

construction project components. Such data representation technique allows great flexibility in project modeling which is discussed in greater detail in the following discussion of GIS-based integrated project modeling framework.

Thirdly, GIS provides the mechanism to effectively streamline many of the information integration tasks such as open data structure, linkage between spatial and non-spatial data, and integral databases. Furthermore, the GIS-based integrated system offers a spectrum of functions and embodies a set of principles and procedures for the collection, storage, management, retrieval, conversion, analysis modeling and visualization of construction data. Visualization and manipulation of an integrated, dynamic project database can provide the user with effective tools to support the comprehension of the complex datasets. In particular, the GIS-based integrated system combines the powerful, sensory-rich presentation capabilities of multimedia information-base with visualization, manipulation and reasoning abilities. The system could serve as an intelligent assistant to project personnel in problem solving in the course of their daily work, or be able to predict problems that might have significant consequences in terms of cost, schedule, quality and safety.

Further support for using GIS as an information integration tool for construction comes from the characteristics of GIS system configuration. GIS is a combination of several different software types: CAD, graphical, database and knowledge-based systems. The common use of CAD, database and knowledge-based systems in construction makes GIS a perfect medium for construction industry. The real strength of GIS occurs when a relational database is linked to graphics. The convenient graphical interface and its strength in database and knowledge base functions make GIS even more desirable than

traditional CAD systems and database systems. Within this technology there are benefits achieved through the development of data and knowledge bases linked to integrated analysis of various construction criteria. Reasoning using standard Knowledge Based Expert System techniques is possible, such as allowing logical queries to be made that link both geometry and other data.

These characteristics meet the research challenge of designing an integrated information system for construction industry. In summary, GIS provides a consistent, coordinate and well-represented communicative platform to handle the complexity and vast amounts of information involved in construction project management. Also it is a medium that enables the examination of interactive procedures as well as the impact of decisions in construction management.

GIS -based Integrated Project Modeling Framework

GIS technology provides a significant platform for construction information integration. Problems remain with the search for the optimal schema for project data handling, especially where adequate transformations of data within a GIS environment in order to feed the models directly are absolutely necessary. Fundamental to this research work is the development of an integrated information or project model that represents the information requirements for construction project information management. A basic role of proposed project modeling framework is to provide a framework with which data can be manipulated and analyzed. The principal activity of any information integration is the gathering and processing of information and the communication of that information to all concerned. A comprehensive information model is proposed to 'anchor' such elements of

information in a unique and systematic fashion so that the interaction can be clearly identified. Essentially, a means of incorporating information integration modeling into the GIS tools are a tremendous aid for such a purpose.

This study has presented an overall view of a rapid prototyping project model. It must be clarified that the proposed integrated modeling framework is not an effort of developing comprehensive project models. The proposed project modeling framework does not include everything on construction project site but uses only those items that are important to demonstrate the effectiveness of GIS-based modeling framework for the representation of project information.

An important feature of proposed project modeling framework is feature objects. Feature objects focus on spatial entities and relationships, and the linkage of non-spatial or attribute data to spatial information describing real world features. The feature object representation strongly promotes the role of spatial data and emphasizes the importance of spatial data representation. Feature objects also emphasize the spatial relationships among the objects being mapped. While a CAD system may represent a wall simply as a line, feature objects in GIS-based integrated information system may also recognize that the wall as the border of the floor. Given the spatial character of construction project data, the use of spatial-lead data representation serves as a good integration medium by providing the basic template for capturing project data.

Not only does it promote the spatial-oriented data integration technique, feature object captures the link between spatial data and non-spatial data that provide a solid ground for the purpose of construction information integration. Data restructuring of feature objects and view objects can be used to analyze construction project information.

Also because of the rich semantics in feature objects and view objects, multiple scenarios of construction project site condition can be evaluated efficiently and effectively. It can provide visual presentation of the exact status of the project, with which user can explore, interact with, and examine interactively the project from multiple perspectives such as material usage, as-built condition and process monitoring.

Another important feature of project modeling framework is the presence of a dynamic representation of reality, the view objects. The proposed data model allows the construction view objects according to measurements of some of their functional requirements and characteristics. As a result, the information of special interest could be rendered.

In general, the challenge for modeling construction project information has been met in the GIS based integrated system in the following ways:

The role of the integrated project-modeling framework can be seen from examination:

- The model strongly promotes the role of information integration scheme in the construction environment by developing and testing the concept of automated acquisition and storage of data in support of project management.
- The model provides an information framework to achieve an effective and coordinated communication of information within the construction domain.
- The model serves as an integrating layer in the integrated information system that provides a means to handle the complexity and the vast amounts of information involved in project management.

- The model is also a medium that enables customized procedures and operations not possible with existing GIS system.

The proposed modeling framework provides a starting point for the application of more sophisticated methods in later work. Methods used in this research suggested improving the efficiency of the linkage between GIS and models include implementing various database management strategies and interfaces within GIS environment. In this research, the use of the object-oriented paradigm within GIS provides a flexible and extensible development environment. The modeling capabilities are improved because of the semantic richness of object-oriented modeling languages. By doing so, the geo-relational model is enhanced by object-oriented operations and suited to the construction project modeling concepts.

The modeling framework is necessarily be the basis for GIS-based integrated system; simply that they are essential for effectively sharing integrated construction project information. But it is not arguing the modeling framework proposed in this research is essential for future development. The modeling framework developed in this research is a basic demonstrates of the research methodology. It does not, however, provide a very rigid framework within which modeling can take place. A more rigid framework might better define the scope of new perspectives. The question of whether the proposed modeling framework remit will be widened in the future remains open. But one of the most important aspects of the modeling framework deals with schema extensibility. The GIS-based model is testified to be possible to implement a functional project database.

GIS-based Integrated Information System Development

Same as the proposed GIS-based modeling framework, the developed application protocol serves as an example how GIS technology can be applied in construction project information integration. The resulted GIS-based integrated information system is a mechanical system for the information analysis and manipulation. Basic function of the GIS-based integrate system is the storage, retrieval, analysis and display of spatial and non-spatial construction project data.

The prototype system strongly promotes the concept of automated acquisition and storage of data in support of project management. Instant spatial data capture provides fast, accurate visual information about the progress on the site. Integration of the spatial database with project management functions provides a powerful and effective management control system. As demonstrated in the developed prototype system, integrated construction project data can be visualized by instantly color-coding the spatial display based on non-spatial attributes such as status. This makes the project data available in a much more powerful and understandable way. Through GIS visualization and multimedia capabilities and easy access to the integrated project data, this integrated system can be used very effectively for manipulation and comprehension of integrated, dynamic project information pool.

In the practical test, the developed system is used to support efficient management of spatial data in many project management applications. Construction project information in a GIS format gives an opportunity to easily query, summarize and organize needed information in any fashion desired. It serves to underpin routine

operations, such as making inventories and monitoring changes, but also support product and process reviews. Preliminary results indicate that considerable potential does exist for use of this system for construction project information management. Various applications have been tested, such as construction activity planning analysis, preliminary cost analysis, as-built schedule analysis and construction project progress analysis.

The value of the GIS-based integrated system development is visible from examination as follows:

- The integrated information system could access, manage and manipulate data in diverse formats and provide information at different level of operation when and where it's needed and its availability for decision-making.
- The system could provide the ability for a great amount of information and have a mechanism built-in for data that represent its correctness, completeness, consistency and non-redundancy.
- The system could present and summarize the project information at multiple detail levels and support the complicated interdependencies of construction information.
- The system would be able to analyze the potential problems and support decision-making.
- The system is technological efficient and effectiveness as well as flexible in processing data.

In general, the GIS-based integrated system for construction information integration is built on sound theoretical foundations. The results of using GIS-based integrated project-modeling framework defining the construction project process and

product and implementing a generic project model in a prototype integrated system development testify the efficiency of the research hypothesis in that:

1. GIS provides a good conceptual framework for building data integration mechanisms.
2. GIS-based modeling framework is effective to be applied to the construction information integration domain.
3. It is possible to supply a set of procedures and operations making it possible to build interfaces between the construction project model and existing GIS packages.
4. The prototype on the project level tests has proved that developing GIS-based integrated system as presented in this research could be possible.

The research demonstrated the adequate functions of GIS-based integrated information systems for getting relevant product data technology in place to effectively support construction project information integration. Methods used in this research also suggested improving the efficiency of the linkage between GIS and models include implementing various database management strategies and interfaces within a GIS environment.

Research Assessment

This research focuses on demonstrating the potential of a GIS (Geographic Information System) in construction project information integration schemes and developing methodology to apply the state-of-art GIS technology and build an integrated

framework and system for construction project information integration. This research situates itself in the translation range between technology promotion and practice. It makes a thorough investigation of GIS-based information integration approach for construction project management, which can be absorbed in practice and provide support on tactical level. It makes the exploration of necessary features of GIS-based integrated system for construction project information management. The approach presented, while being original, also has great practical potential; it is hoped that its adoption will lead to an improvement in the quality of the construction project information integration.

The limitation of the research methodology lies in its scope and its perspective. The objectives stated at the outset of the project are very challenging. Unfortunately this proved to be too ambitious when taking account research resources. The research approach identifies GIS technology perspectives for construction information integration at a relatively preliminary level. A GIS-based integrated information system for the construction industry is still in a very early phase of development. The size of the industry being modeled means that larger shared and commonly understood domains are not recognized and integrated until a later stage of development. There are issues that need to be discussed, researched, tested in order to implement a fully integrated system that provides information from pre-design to maintenance and demolition. Further testing and refinement of the system is required to determine its viability in real practice. This has implications for the completeness and correctness of such research project. But the research methodology does not constrain research perspectives in that they may settle information integration needs.

This research focused on the still narrow domain and limited business process. There is a variety of information involved in construction site management. This research so far targets integration at a relatively coarse grain size. No attempt has been made to clearly define a project setting, e.g., participating actors involved in a specific task. As a result of this, the resulting suite of tools/actors represents a more or less random selection in a moreover unspecified construction process and project environment.

Also, there are a number of computational issues to be dealt with in developing such a system concerning search and retrieval capabilities, security to prevent unauthorized access, storage management and document tracking. At this point, research focus is on the organization of information to support efficient search and retrieval, while realized that other issues need to be addressed. Finally, the current implementation performance is not adequate for commercial use and would require optimization.

This research concentrated on data integration. This research showed how the data integration works through the GIS environment and construction project modeling. On operational level moderate goals were reached, in the sense that the demo showed the set of modules, where the data integration operations is implemented in order to support cooperation among actors, engaged in the context of simulated project. No consideration is given to the construction data flow process itself. No data exchange with existing construction application softwares are considered. Even more out of scope is the cooperation between actors engaged in construction project. No attempt is done to supply support on the control over actors' cooperation. Functionality on flexible and intelligent and eventually concurrent multi-actor control was not offered. It should be clear that the

resulting research deliverables is exclusively on data-integration level, thus is inherently very limited in their potential of proving dynamic information integration support.

Future Development

After the assessment of the research deliverables, a number of obvious extensions are identified. Among them the scope extension is necessary by taking a broader range of data and tools into account that are vital to arrive at usable system. The integrated building model need to be expanded and in fact re-tailored considerably to reach a much richer level. There is also a great need to re-tailor the GIS-based information-modeling framework for better maintainability and expandability and enrich the project model in key areas.

For future stages of research in general, it is readily seen that the main long-term challenges concern increased levels of semantic coverage and their conceptual modelling tool support. On the implementation side it is opted to steer towards robust implementation of the project model with persistence support in an object-oriented database (OODB) and a re-engineered interface kit with enriched functionality. Apart from these scope-extensions a number of the additional project management tools necessitate a much richer integration among heterogeneous applications. The inclusion of on-line tools is another major new direction in the research. Adequate data exchange tools should be provided. Other follow-ups will have to deal with the configuration of the new systems in the management settings of the co-operating companies, taking account of changing organisational structures induced by the co-operative engineering technology.

It is recommended that future work in this area should investigate:

- To enable the use of representation strategies for improving the performance of modeling framework and system implementation;
- To enable the use of concurrent hardware/software technologies as enabling and intelligent aids;
- To apply industry standard project models;
- To a large extent, provide an open and extensible architecture and support the integration of a large population of existing and new tools for construction management.
- Much further development work is needed to get client server based component database for wide use in practice.

Another major extension steered towards delivering the system to real life project settings. This will predominantly address issues on tactical level by focusing on a number of project environments, identifying the actors in various scenarios and configuring the exchange system accordingly. A primary issue is the enterprise-oriented support within the system, which could span the whole of the project (as far as the enterprise is concerned) or only part of it. The multi-actor process requires adequate support for collaborative group work procedures. These systems will break through the barrier of limited data exchange by adding means to control the exchange events in chosen scenarios. Enterprise integration of tools on strategic level will be the ultimate challenge, as it requires on top of the previous. Enterprise-wide modelling environments in which integrated system can be embedded, guaranteeing full control, management and updating facilities.

The future development in this area could be to:

- Support the integration of the project team, the project evolution process, and the project lifecycle activities.
- Support the necessary transactions, collaboration/negotiation and communication between the project team.
- Accommodate a distributed computing environment with appropriate levels of security, based on open networking architecture to allow teams to work transparently.
- Provide facilities for the generation, representation and capture of knowledge within a particular domain, support the efficient use of previous knowledge, methods and procedures.

The GIS-based integrated system research is making an effort to develop new integrated information systems that have a credible potential to be absorbed into practice. Involvement of practice through benchmarking in the work place is essential to reach long-term objectives. It is anticipated that follow-ups will include benchmark tests on real life projects in multi-actor and multi-partner environments. To the end user community the message lies in the outcome of the down stream field testing which should generate enough market pull for this type of integrated systems. It is because of this that integration needs have to be addressed in an overall enterprise-wide management context, whereas inevitable modifications and project-specific tailoring of tools required tremendous input from industry professionals.

What will the future bring? It is clear that we will see a gradual development from simple prototype systems to more sophisticated systems as envisioned.

APPENDIX A

PROJECT OBJECT LIBRARY SPECIFICATION

The construction project feature object model is implemented as an ODB (Object Data Base). An ODB provides a file-based storage and retrieval system for feature objects. In the ODB, project models are defined as feature objects. An Avenue class defines a common set of attributes and operations of a feature object class. The structure of the implementation could be found in figure (AA-1).

An ArcView feature theme (*FTheme*) contains the spatial entity. The spatial entity is defined as a *Class* that must be one of *Point*, *PolyLine*, *Polygon*, or *MultiPoint*. As the spatial entity is extracted from CAD drawing, most of the primary spatial entity is represented as either point or line, which could be transformed to shape class by different methods. The symbol of the spatial representation is specified legend according to the definition of color, size, marker, line style, fill style. It is also possible to specify related multimedia information including daily reports in the form of images and video. The behavior of the geometric shape is defined as *Shape*.

A feature table (*FTab*) stores the property set associated with the feature object definition. Objects are initialized to the defaults of their respective feature table. All the data about entities and their spatial and non-spatial attributes will be stored in a FTAB as *Fields*. Each value in a FTAB can be accessed with a record number of a *Field*. A *Field* defines what kind of values can be stored in the field. The *Field* type for the definition of

feature object attribute uses some of the defined field type in ArcView and some added definition by the author.

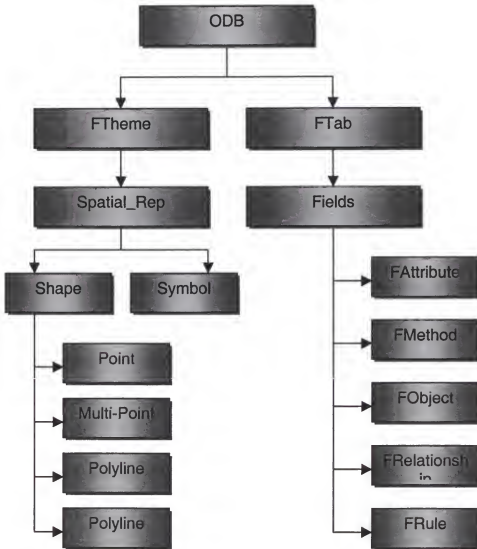


Figure AA-1: Implementation structure

The defined complex field types of project feature object are as the following:

- *FShape*: the shape class of the spatial entity.

- *FAttribute*: the value of the specific attribute of special type such as number, string etc.
- *FObject*: name of feature object from which the attribute links with.
- *PREFERENCE*: name of object and attributes from which the value of field is referred to.
- *FMETHOD*: name of the script that running for calculating the value of the field and the return.
- *FRELATIONSHIP*: the name of the script that build relationship and the input.

The relationships of feature object are attached to the *FTab* as a field with the relationship defined and the related object. The relationship is defined by a script and the related object is get from user input. Once these relationships are established, the properties of the related objects affected by the relation are updated. Once the relationship is built, object can request information from its related object, the properties of the related objects included the relation are then send back to the requesting object. Rules are developed using typical (if→ then) format and are triggered by pattern matching. Once the status of the *FTab* or *FTheme* is changed, the rules are checked and if fired the rule script runs. The status of feature object is defined in two categories: new or old. The version is documented with database time stamp that is automatically recorded every time a new version is initiated.

All project object instances in the project central database are created using the templates defined in the form of their respective classes. One desirable method for each project information object is to *create a version* of them, and maintain the history of

themselves. Then the objects sent messages so that they may relate to each other. Once the relationship is built, object can request information from its related object, the properties of the related objects included the relation are then send back to the requesting object. The rules for establishing the relationships and computing the updated values are stored in the respective classes. Also it is a desirable method for an object to be able to notify the system of changes once the value of itself or its attributes has been modified. On other condition, once object received the notification from system, the value of affected attributes will be update automatically.

Project Feature Object Library

The project feature object library in its current form consists of eight categories of objects. These are product objects, activity objects, method objects, resource objects, specification objects, participant objects and feedback objects. Due to the space restrictions, at this level, the main attributes for each of these classes are presented. More specific attributes and subtypes would be defined in lower-level subclasses. And only the abstract definitions of the classes have been presented, and detail attributes and methods have been omitted for simplicity and clarity.

These classes are not intended to be mutually exclusive. For example, a foreman is a participant of a construction activity but should also be counted as a human resource since his time will be charged to the activity (we expect most activity costs to arise through resource utilization issues). Rather than try to fit this possibility into the definitions of either the Participant or Resource classes, we assume that a class used to represent Foremen would be a subtype of both Participant and Resource.

Product Feature Objects

- **Definition:** Product objects are systems and components of the constructed facility itself. The product object is the first class object. That is, when a project start, the product object creation is required and the product object initiate two other project objects, activity object and resource object (specifically, material object).

- **Hierarchical Structure:** Conceptually, a product object is broken down in a hierarchy based on a chosen level of elements. The element is in someway a product object but goes one level deeper. For element attribute, the same set of attributes is applied. Sub-types of products include wall, floors, roofs, utility and etc. The Subtype depends upon template object definition the subtype inherit their definition. The elements and subtypes are defined when the object is instantiated.

- **Spatial-Rep:** Spatial-Rep of product object is transformed from CAD drawing by specifying layer. The shape class of the product object is one of the four shape classes, point, multi-point, polyline, or polygon respectively. This spatial-rep is shared among the related objects.

- **Characteristics attributes:** The characteristic attributes for class, object ID, type, quantity, dimension, material (intended, built), and drawing item (shop drawing, as-built drawing). The class is the select from an enumeration kit of feature objects defined in object library. The product object is thus declared to be a certain feature object. The element ID is a user input integer number attached to the object enumeration, such as 'wall-2'. The element type is the select subtype from an enumeration kit defined according to the object declared. For example, an element type for a declared 'roof' object

is select from 'metal roof', 'shingle roof' etc. The location is 1D spatial key to the feature object. The quantity is the area, length, count for polygon, polyline and multipoint classes respectively. The dimension is the user input of the other dimension not included in the spatial representative such as thickness of a line feature, or size of a point feature. The material is the resource intended or used. Drawing items can be categorized portions of drawing files by specifying the drawing file name, layers and entity information.

- Relationships: The product object has the *compose_of* relationship with other product object to represent the assembly of a building product, and inherits attributes and behaviors based on component and material types. The product object also has the *connected_to* relationship with other product object. The connected-to relationship defines the spatial supporting between product objects. At the same time, the product state object has the conflict-with relationship with other product objects to define the spatial conflict. For example, the drinking fountain has the *connect_to* relationship with water pipeline, and conflict relationship with door. These two relationships require specific spatial range. For example, conflict with another product object within 10 feet, connect to another object within 0 foot. The default *connect_to* and *conflict_with* relationships are stored. The product object also has the constraint relationship with activity object and material object.

- Rules: If the product object is created, accordingly create related activity object, method object and material object. The product object passes the constraints of itself to the initiated object classes. If the product object is edited, update attributes and relationships. If the quantity, method and material attributes are changed, inform changes

to activity object, method object and resource object. If the product object is deleted, accordingly delete the related activity object and update the relationships.

- Database Directives:

FIELD Class: *FObject* = *SELECT* <one of product object classes>

FIELD ID: *FAttribute* = *INTEGER*

FIELD Type: *FAttribute* = *SELECT* <one of enumeration>

FIELD Measurement: *FAttribute* = *SELECT* <one of enumeration>

FIELD Quantity: *FMethod* = &av.Run("Srp_getQuantity")

FIELD Material: *FObject* = *SELECT* <set of resource object classes>

FIELD composed_of : *FRelationship* = &av.Run

("Srp_composedRel", {<property>, <set of product object>})

FIELD connect_to : *FRelationship* = &av.Run

({"Srp_connectRel", {<range>, <set of product object>})

FIELD conflict_with: *Frelationship* = &av.Run

({"Srp_conflictRel", {<range>, <set of product object>})

FIELD Picture Item (Hot Field): *filename*

Activity Feature Objects

- Definition: Activity objects represent processes or actual activity and construction effort on the project. Processes are performed by actors, they apply resources and process input products, they have proceeding and succeeding processes, and they result in other product objects. The activity object is the second class object. It is

initiated by first class of product object. And it will initiate two third class objects of participant object and resource object.

- **Hierarchical Structure:** The zone, assembly, and activity classification hierarchies are used to create the activity packages. The activity breakdown structure is related to the definition of related product breakdown structure. The definition of activity object at lowest level is equivalent to assembly of components belonging to a particular product object. The activities are grouped together by zone to form a work package. Activity objects inherit their work element and their possible team compositions and dependencies relationship. The activity object is the second-class object that means they are created by a first class object, i.e., product object. The activity also initiates a resource object (crew and equipment) and method object.

- **Spatial-Rep:** Activities reference their spatial-Rep from the work elements in which they are executed.

- **Characteristic Attributes:** Their characteristic attributes include class, ID, work elements, subcontractor, method, and process. Class attribute is a user selection from a list of activity object list which is initiated by product objects. Because the activity object is second-class object and is initiated by product object, when an activity object is created, the product object that initiate the activity object is stored as a constraint. The process is the percentage completed. Work elements are a list of product the activity will work on.

- **Relationships:** The *activity* object has the *preceded_by* and *succeeded_by* relationships with the other activity object. The *preceded_by* and *succeeded_by* relationships represent the constraint of one or more activities in terms of start to finish.

For example, an activity object that is prerequisite for the start of the work “floor tiling” is “completion of the floor slab”, which is the end condition of the work “floor leveling”. Also the activity has the constraint relationship with product object. This relationship is defined through the attribute of state.

- Rules: If the activity object is created, initiate the according method object and resource object. If the activity object is modified, updates relationships and values of related resource object and participant object. If the state of activity object is changed, inform the product object.

- Database Directives:

FIELD Class: *FObject* = *SELECT* <one of activity object classes>

FIELD ID: *FAttributet* = *INTEGER*

FIELD Work_Package: *FObject* = *List*

FIELD Subcontractor: *FAttributet* = *STRING*

FIELD Process: *FAttributet* = *SELECT* <one of enumeration>

FIELD precede_by : *FRelationship* = &av.Run

({"Srp_precedenceRel", {<precede>, <set of activity object>})

FIELD succeed_by: *FRelationship* = &av.Run

({"Srp_precedenceRel", {<succeed>, <set of activity object>})

Resource Feature Objects

- Definition: Resource objects represent the resources that is made use of in performing activities and used in carrying out product. The resource object is a third class

object. The material resource object is initiated by product object. The labor and equipment resource object is initiated by activity object.

- Hierarchical Structure: Subtypes of resource object include labor force, material and equipment, or compositions of these three basic construction resources.

- Spatial-Rep: Resource objects reference their spatial-Rep from the product/activity object in which they are used.

- Characteristic attributes: For construction resources, the characteristic properties are class, resource type, unit of measure, unit cost and provider. Class attribute is a user selection from material, equipment or labor. The resource type for material class is a list of resource object list that is initiated by product objects. The resource type for equipment and labor classes are initiated by activity objects. Optionally, the resource object could have a picture item to display material or equipment.

- Rules: when a resource object is created, an associated participant feature object is initiated.

- Database Directives:

FIELD Class: *FObject* = *SELECT* <one of enumeration>

FIELD Type: *FObject* = *SELECT* <one of resource feature objects>

FIELD Quantity: *FAttribute* = *REAL*

FIELD Unit of Measure: *FAttribute* = *STRING*

FIELD Unit of Cost: *FAttribute* = *MONEY*

FIELD Provider: *FObject* = <participant object>

FIELD Description: *FAttribute* = *STRING*

FIELD Picture Item (Hot Field): *filename*

Method Feature Objects

- Definition: Method objects describe the technique or methodology used to perform the activity and construct the product. The method object is the third level class, it is initiated when an activity object is created. The method object is also a knowledge object, which means it just delivers the knowledge of a certain method. The construction method provides the productivity data required for the calculation of activity duration.

- Hierarchical Structure: It has only one level of object.
- Characteristic Attributes: For each type of equipment and crew used for an activity, a unique construction method object is defined. Thus, every construction method object has equipment and labor attributes. The team composition is defined as a combination of different type of labor or equipment used on an activity, and the quantity for each type of labor or equipment used. There are five characteristic attributes defined in this class: activity, method, labor, equipment and productivity. The activity attribute is the class of activity that uses the method. Productivity is the user defined rate for specific team composition.

- Database Directive:

FIELD Activity: *FObject* = *SELECT* <one of activity object classes>

FIELD Method: *FObject* = *SELECT* <one of method object classes>

FIELD Equipment: *FAttribute* = *LIST*

FIELD Labor: *FAttribute* = *LIST*

FIELD Productivity: *FAttribute* = *REAL*

Participant Feature Object

- Definition: The participant object represents the professional groups in the project.
- Hierarchical structure: The participant object has the sub class of subcontractors, material supporter, equipment renter and etc.
- Characteristic Attributes: The characteristic attributes of participant object include class, participant, contact info, contract scope, contract type and contract amount.
- Database directives:

FIELD Class: *FObject = SELECT <one of participant object classes>*

FIELD Participant: *FObject = SELECT <one of participant objects>*

FIELD Address: *FAttribute = STRING*

FIELD Phone: *FAttribute = STRING*

FIELD Fax: *FAttribute = STRING*

FIELD Email: *FAttribute = STRING*

FIELD Contract Type: *FAttribute = SELECT <set of enumeration>*

FIELD Contract Scope: *FAttribute = SELECT <set of enumeration>*

FIELD Contract Amount: *FAttribute = MONEY*

Project View Object Library

In the project view object library, three view objects are defined: cost view, site view and process view.

Cost view object deals with the cost structure of individual parts as well as the overall project from various perspectives (sub-trades, general, owner). Cost domain object contains the declarative and procedural knowledge that assists user in performing elemental cost estimating and cost tracking throughout the construction phase. The class methods not only establish a unified view to get messages (quantity & quantity) from building projects, but also to perform an elemental cost estimate; and display the analysis results to the view.

Site view object describes the performance of how a project was built physically. This view would include information such as an identification geometry, functionality of building spaces, basic building materials, etc. The class methods establish a unified view to browse stored product information, to perform product analysis and display the results to the view. Also it assists in progress monitoring and performance and process monitoring. The view methods not only establish a unified view to browse stored information, but also to analyze the project performance and display the analysis results to the view.

Process view object describes how the project will be constructed, who is responsible for different aspects of the work, when it will be done, and where. Process view object contains the declarative and procedural knowledge that assists user in performing project planning and scheduling. This object contains methods that not only establish a unified view to browse project plan, but also to analyze construction activity; perform an elemental scheduling; and display the analysis results to the view

These view objects are described in greater detail in the following pages. The description of these view objects includes the definition of the view object, characteristic attributes of view object, spatial representation and database directives.

Cost View Object

A. Definition: The cost view object represents the information structure of cost estimating domain objects using the subset of related objects as its instance variables.

B. Characteristic Attributes: The characteristic attributes for cost view objects include:

- Element Item: is a subset of the product object, and is used to store the attributes of element type.
- Work Item: is a subset of the activity object, and is used to store the attributes of work that cost item will be applied.
- Element Quantity: is a subset of the product object, and is used to store the attribute of element quantity.
- Unit Cost: is a subset of the resource object, and is used to store the unit cost of certain material.
- Provider: is a subset of the participant object, and is used to store the provider attribute.
- Cost (planned or actual): is used to store the itemized planned or actual cost of selected assembly elements and work elements.

C. Spatial-Rep: Cost view objects reference their spatial-Rep from the center point of product elements in which they are applied.

D. Database directives:

FIELD Resource: *FReference*= REFERENCE <object> of <aResourceObj>

FIELD Element Item: *FReference*= REFERENCE <type> of <aProductObj>

FIELD Element Quantity: *FReference*= REFERENCE <quantity> of <aProductObj>

FIELD Work : *FReference*= REFERENCE <object> of <aActivityObj>

FIELD Unit: *FReference*= REFERENCE <unit_of_measure> of <aResourceObj>

FIELD Unit Cost: *FReference*= REFERENCE <unit cost> of <aResourceObj>

FIELD Provider: *FReference*= REFERENCE <provider> of <aResourceObj>

FIELD Picture Item: *FReference*= REFERENCE <picture item> of <aResourceObj>

Site View Object

A. Definition: The site view object represents the complex information structure of product control domain objects using the objects of the following classes as its instance variables. The definition of the class is listed below.

B. Characteristic Attributes: The characteristic class attributes is listed below:

- Element: is a subset of product object, and is used to store the element attribute.

- **Type:** is a subset of product object, and is used to store the element type attribute.

- **Material:** is a subset of product object, and is used to store the material type attribute.

- **Quantity:** is a subset of the product object, and is used to store the attribute of element quantity.

- **Drawing Item:** is a subset of product object, and is used to store the drawing item.

- **Work:** is a subset of activity object, and is used to store the work attribute.

- **Subcontractor:** is a subset of activity object, and is used to store the subcontractor attribute.

- **Methods:** is a subset of activity object, and is used to store the method attribute.

- **Percentage Completed:** is used to store the percentage of product construction completed, and is referred the process attribute of activity object.

- **Connect_to:** is a subset of product object, and is used to store the connection relationship.

- **Conflict_with:** is a subset of product object, and is used to store the confliction relationship.

C. **Spatial-Rep:** Site view objects reference their spatial-Rep as the exact shape of product elements in which they are applied.

D. **Database directive:**

FIELD Element: *FReference* = *REFERENCE* <object> of <aProductObj>

FIELD Quantity: *FReference*= *REFERENCE* <quantity> of <aProductObj>

FIELD Material: *FReference*= *REFERENCE* <material> of <aProductObj>

FIELD Drawing Item: *FReference*= *REFERENCE* <drawingItem> of
<aProductObj>

FIELD Work: *FReference*= *REFERENCE* <object> of <aActivityObj>

FIELD Subcontractor: *FReference*= *REFERENCE* <subcontractor> of
<aActivityObj>

FIELD Methods: *FReference*= *REFERENCE* <method> of <aActivityObj>

FIELD Process: *FReference*= *REFERENCE* <process> of <aActivityObj>

FIELD Compose_of: *FReference*= *REFERENCE*
<compose_of> of <aProductObj>

FIELD Connect_to: *FReference*= *REFERENCE*
<connect_to> of <aProductObj>

FIELD Conflict_with: *FReference*= *REFERENCE*
<conflict_with> of <aProductObj>

Process View Object

A. Class definition: The Process view object represents the complex information structure of process planning and scheduling domain objects using the objects of the following classes as its instance variables.

B. Characteristic Attributes: The characteristic attributes for cost view objects include:

- Work: is a subset of activity object, and is used to store the attributes of activity type.
- Work Element: is a subset of the activity object, and is used to store the attribute of element.
- Work Quantity: is a subset of the product object, and is used to store the attribute of quantity.
- Material: is a subset of the product object, and is used to store the attribute of material.
- Crew: is a subset of the method object, and is used to store the attribute of crew attribute.
- Equipment: is a subset of the method object, and is used to store the attribute of equipment.
- Productivity: is a subset of the method object, and is used to store the attribute of productivity.
- Process: is a subset of activity object, and is used to store the attribute of process.
- Proceed_by: is a subset of activity object, and is used to store the precedence relationship.
- Succeed_by: is a subset of activity object, and is used to store the succedence relationship.
- Planed Duration: is used to store the planned duration of certain activity, and is a calculation result of quantity, team composition and rate which is referred to the knowledge of related product object, activity object and method object.

- **Actual Duration:** is used to store the actual planned duration of certain activity, and is the calculation of actual start time and actual finish time.

- **Planned Start Time:** is used to store the planned start time of the activity, and is the calculation result of activity precedence and timing.

- **Planned End Time:** is used to store the planned end time of the activity, and is the calculation result of activity precedence, timing and planned duration.

- **As-built Start Time:** is used to store the as-built start time of the activity, and is referred to a date of specific version of activity object.

- **As-built End Time:** is used to store the as-built end time of the activity, and is referred to a date of specific version of activity object.

- **Start Lag State:** is used to store the start timing lag of certain activity, and is the comparison result of planned start time and as-built start time. The value of this attribute could be one of three: late start, scheduled start or early start.

- **End Lag State:** is used to store the end timing lag of certain activity, and is the comparison result of planned end time and as-built end time. The value of this attribute could be one of three: late end, scheduled end or early end.

C. **Spatial-Rep:** Site view objects reference their spatial-Rep as the extent of product elements in which they are executed.

D. **Database directive:**

FIELD Work: *FReference*= *REFERENCE* <object> of <aActivityObj>

FIELD Work Element: *FReference*= *REFERENCE* <work elementt> of <aActivityObj>

FIELD Work Quantity: *FReference*= *REFERENCE* <quantity> of
<*aProductObj*>

FIELD Material: *FReference*= *REFERENCE* <material> of <*aProductObj*>

FIELD Method: *FReference*= *REFERENCE* <method> of <*aActivityObj*>

FIELD Crew: *FReference*= *REFERENCE* <crew> of <*aMethodObj*>

FIELD Equipment: *FReference*= *REFERENCE* <equipment> of <*aMethodObj*>

FIELD Productivity: *FReference*= *REFERENCE* <productivity> of
<*aMethodObj*>

FIELD Process: *FReference*= *REFERENCE* <process> of <*aActivityObj*>

FIELD Proceed_by: *FReference*= *REFERENCE* <proceed_by> of
<*aActivityObj*>

FIELD Succeed_by: *FReference*= *REFERENCE* <succeed_by> of
<*aActivityObj*>

FIELD Planned Start Time: *FMethod*= &av.Run("ProcessView.GenSchedule")

FIELD Planned End Time: *FMethod*= &av.Run("ProcessView.GenSchedule")

FIELD As-built Start Time: *FMethod*= &av.Run("ProcessView.GenABSchedule")

FIELD As-built End Time: *FMethod*= &av.Run("ProcessView.GenABSchedule")

FIELD Planned Duration: *FMethod*= &av.Run("ProcessView.DefineItem")

FIELD Actual Duration: *FMethod*= &av.Run("ProcessView.GenABSchedule")

FIELD Start Lag State: *FMethod*= &av.Run("ProcessView.CheckStatus")

FIELD End Lag State: *FMethod*= &av.Run("ProcessView.CheckStatus")

The implementation of the construction project feature object model is implemented using Avenue coding. Avenue is the programming language that comes with ESRI ArcView GIS.

APPENDIX B

CONSTRUCTION GIS SYSTEM DESCRIPTION

In this section, the description for Construction GIS system is included. The description is used as a user manual for the system interface. It includes explanation of functions of window displays, buttons, tools, menus and dialogs, and how to use these controls.

Project Window

- A. Purpose: Manipulate project files and navigate functional windows.
- B. Composition: This unit combines 1 project portfolio display and 3 menus.
- C. Portfolio display: The project portfolio display contains all the relevant documents available in the project, including maps, database tables and charts.
- D. Menu: Three menus are available for this window:
 - 1: Project menu including 4 menu items:
 - Set Working Directory: is used to set working directory to a certain drive and directory;
 - Save Project: is used to save project work;
 - Close Project: is used to close project file;
 - Exit: is used to exit Construction GIS application.
 - 2: Window menu including 5 menu items:

- CAD Definition Window: is used to open CAD Definition Window;
- Feature Object Definition Window: is used to open Feature Object

Definition Window;

- Cost View Window: is used to open Cost View Window;
- Process View Window: is used to open Process View Window;
- Site View Window: is used to open Site View Window.

3: Help menu including 2 menu items:

- About Construction GIS: is used to open documentation on Construction

GIS system description;

- Construction GIS User Manual: is used to open Construction GIS system

user manual documentation.

CAD Object Definition Window

A. Purpose: Define the CAD drawing items to meaningful spatial objects.

B. Composition: This unit combines 1 display view, 4 menus and 12 tools and a dialog serving to enter the object definition value.

C. Display view: The view displays imported CAD drawing and defined spatial object themes.

D. Menu: Four menus are available for this unit:

1: File menu: this menu includes 7 menu items:

- Add CAD: is used to add CAD .dxf file to the view;

- Delete CAD: is used to delete the user-selected CAD drawing from the view;

- Add Theme: is used to add the defined CAD theme, e.g. wall theme, floor theme, etc., to the view;

- Delete Theme: is used to delete user selected defined theme from the view
- Print: is used to print the view;
- Export: is used to export user selected theme to an output file;
- Close: is used to close the CAD Definition Window;

2. CAD Object Definition menu: this menu includes 4 items:

- Define CAD Objects: is used to activate CAD object definition tools;
- Edit CAD objects: is used to activate CAD object editing tools;
- Save Edits: is used to save the editing of the CAD object, this tool is disabled if there is nothing in editing mode;

- Show Tables: is used to open database tables associated with CAD Object Definition Window.

3. Graphics menu: this menu includes 3 items:

- Edit Legend: is used to open legend editing dialog to edit legend for the active theme;

- Label Theme: is used to open label dialog to label the all records of the active theme;

- Clear Graphics: is used to clear graphics on the map, including labels and graphic shapes;

- Clear Selection: is used to clear selection on the active theme;

- Zoom to Full Extent: is used to display the map at full extent;
- Zoom to Theme: is used to display the map at the extent of the active

theme;

- Zoom to Previous: is used to display the map at previous zoom scale;
- Zoom In: is used to zoom in the map display;
- Zoom Out: is used to zoom out the map display;

4: Windows menu: this menu includes 6 items:

- CAD Definition Window: is used to close active window and open CAD

Definition Window.

- Feature Object Definition Window: is used to close active window and

open Feature Object Definition Window.

- Site View Window: is used to close active window and open Site View

Window.

- Cost View Window: is used to close active window and open Cost View

Window.

- Process View Window: is used to close active window and open Process

View Window.

- Project Portfolio Window: is used to close active window and open

Project Portfolio Window.

4: Help menu: this menu is the same as in Project Window.

E. Tools: Three sets of tools are available for this unit.

1. CAD Definition Tool Set: This tool set is used for defining CAD objects. It is activated by user click the menu of “Define CAD Object”. This set includes four tools:

- Spatial object definition tool: is used to enter the definition of user-selected object. User uses mouse to draw a rectangle around the interested object on the CAD drawing. Then a CAD Definition Dialog is shown for user to enter the class, object id and z_value (optional) for the selected object.

- Spatial object edit tool: is used to edit the definition values for user-selected object. User selects the object first. Then the CAD Definition Dialog is shown with the value of class, object id and z_value of selected object shown in the dialog. User could change the value and edit the definition of the selected.

- Object identify tool: is used to identify user-selected object. User first makes the interested theme the active theme and clicks on the interested object. The value of the selected object is shown on the identify window.

- Object label tool: is used to label selected object. User clicks on the interested object. The object then is labeled with the object id.

2. CAD Editing Tool Set: This tool set is used for editing CAD objects. It is activated by user click the menu of “Edit CAD Object”. This set includes four tools:

- Select to edit tool: is used to edit the spatial feature selected by user. User could use this tool to edit the location and shape of the selected feature;

- Select to delete tool: is used to delete the spatial feature selected by user;

- Select point tool: is used to select the spatial feature. User could select the spatial feature and move it to another location;

- Add feature tool: is used to add the spatial feature selected by user. This tool is interactive with the editing theme. If the editing theme is point feature, the tool is adding spatial point tool. If the editing theme is line feature, the tool is adding spatial line tool. If the editing theme is polygon feature, the tool is adding spatial polygon tool.

3. View Navigating Tool Set: This tool set is used to navigate the view. It includes four tools:

- Zoom in tool: is used to zoom in on the position user clicks or the box user defines on a view.

- Zoom out tool: is used to zoom out from the position user click or the area user defines on a view.

- Pan tool: is used to pan the view by user dragging it in any direction with the Pan tool.

- Measure tool: is used to measure distance on a view. Use the mouse to draw a line defining the distance user wants to measure. The measurements are displayed in the ArcView status bar, shown in the current distance units of the view.

F. Dialog: A dialog serves as CAD object entry tools for this unit. In this dialog, there are three entries:

- Class entry: user defines class definition by selecting from a list.
- Id entry: user defines the object id by entering the id number at the input prompt.
- Z value entry: user defines the z value by entering the decimal values. This entry is optional.

The entry value will be saved to database and added to the defined theme if user clicks apply button. User clicks cancel button to dismiss the dialog and no value is entered to database. User could also open on-line help text by clicking the help button.

Feature Object Definition Window

A. Purpose: Define the feature objects including product object, activity object, resource object, method object and participant object.

B. Composition: This unit combines 1 view display, 5 table displays, 4 menus and 8 buttons and 5 dialogs serving to enter the object definition value.

C. Display view: The view displays imported defined spatial object themes. Each table of five displays the feature object database.

D. Menu: Four menus are available for this unit:

1: File menu: same as in CAD Object Definition Window.

2. Feature Object Definition menu: this menu includes 5 menu items:

- Define Product Feature Object: is used to bring up the product feature object definition dialog for user to enter object definition values and save the product object definition value in the product feature object database.

- Define Activity Feature Object: is used to bring up the activity feature object definition dialog for user to enter object definition values and save the activity object definition value in the activity feature object database.

- Define Resource Feature Object: is used to bring up the resource feature object definition dialog for user to enter object definition values and save the resource object definition value in the resource feature object database.

- Define Method Feature Object: is used to bring up the method feature object definition dialog for user to enter object definition values and save the method object definition value in the method feature object database.

- Define Participant Feature Object: is used to bring up the participant feature object definition dialog for user to enter object definition values and save the participant object definition value in the participant feature object database.

3. Graphics menu: same as in CAD Object Definition Window.

4. Window menu: same as in CAD Object Definition Window.

5. Help menu: same as in Project Window.

E. Buttons: Two sets of buttons are available for this unit.

1. Feature Object Definition Buttons: This set of buttons is used for defining feature objects. It includes five buttons:

- Product Feature Object Button: is used to bring up the product feature object definition dialog for user to enter object definition values and save the product object definition value in the product feature object database.

- Activity Feature Object Button: is used to bring up the activity feature object definition dialog for user to enter object definition values and save the activity object definition value in the activity feature object database.

- **Resource Feature Object Button:** is used to bring up the resource feature object definition dialog for user to enter object definition values and save the resource object definition value in the resource feature object database.

- **Method Feature Object Button:** is used to bring up the method feature object definition dialog for user to enter object definition values and save the method object definition value in the method feature object database.

- **Participant Feature Object Button:** is used to bring up the participant feature object definition dialog for user to enter object definition values and save the participant object definition value in the participant feature object database.

2. **View Navigation Buttons:** This set of buttons is used to navigate the view display. It includes five buttons:

- **Zoom to Full Extent Button:** is used to zoom to the full extent of all the themes in a view.

- **Zoom to Theme Button:** is used to zoom to the spatial extent of the active theme(s).

- **Zoom in Button:** is used to zoom in on the center of a view or a layout by a factor of 2.0.

- **Zoom out Button:** is used to zoom out from the center of a view or a layout by a factor of 2.0.

- **Zoom to Previous Button:** is used to go back to the previous extent before zoomed or panned.

F. **Dialogs:** Five dialogs are used by this unit to provide value entries for feature object definition, including:

1. Product Feature Object Definition Dialog: This dialog is used for value entries for product feature object definition. There are 9 entries in the dialog:

- Class entry: user defines class definition by selecting from a list.
- Id entry: user defines the object id by entering the id number at the input prompt. If there is an object with the id number user entered and stored in the product feature object database, the value of that object will fill in the rest of the entries, and the edit button is activated.
- Type entry: user defines the type value by selecting from a list. The available values in the selection list are defined by the class entry user selected.
- Material entry: user defines the type value by selecting from a list. The available values in the selection list are defined by the class entry user selected.
- Measurement entry: user defines the measurement value by selecting from a measurement unit list.
- Picture Item entry: user clicks the browsing button next to it and select a picture item file, the filename and path of that file then is shown at the text line. This value is optional.
- Compose_of entry: user defines the *compose_of* value by three steps. First, user selects a class from a list. The available values in the selection list are defined by the class entry user selected. Then user enters the id number for a certain object of the class. The add button beneath the selection prompt is activated. User click the add button, the *compose_of* object is entered into the *compose_of* object list. User could select a *compose_of* object from the *compose_of* object list. The remove button above the list is activated. User click the remove button, then the selected object is removed from the

compose_of object list. If there is no object in the *compose_of* object list, the remove button will be automatically turned inactive.

- *Connect_to* entry: user defines the *connect_to* value by three steps. First, user selects a class from a list. The available values in the selection list are defined by the class entry user selected. Then user enters the id number for a certain object of the class. The add button beneath the selection prompt is activated. User click the add button, the *connect_to* object is entered into the *connect_to* object list. User could select a *connect_to* object from the *connect_to* object list. The remove button above the list is activated. User click the remove button, then the selected object is removed from the *connect_to* object list. If there is no object in the *connect_to* object list, the remove button will be automatically turned inactive.

- *Conflict_with* entry: user defines the *conflict_with* value by three steps. First, user selects a class from a list. The available values in the selection list are defined by the class entry user selected. Then user enters the id number for a certain object of the class. The add button beneath the selection prompt is activated. User click the add button, the *conflict_with* object is entered into the *conflict_with* object list. User could select a *conflict_with* object from the *conflict_with* object list. The remove button above the list is activated. User click the remove button, then the selected object is removed from the *conflict_with* object list. If there is no object in the *conflict_with* object list, the remove button will be automatically turned inactive.

After necessary entries are entered, user could click new record button. The value for the object user defined will be entered into the product feature object database as a new object or if there are already one or more versions of the object, the values will be

saved to database as a new version of the object. The relationships will be established at the point the object is saved into database. If there is one or more versions of object in the product feature object database, user enter the id number for the object, the values for the latest version of the object will fill in the entries. User could edit the values. And then click the edit record button. The edited value will be saved to the database as the same version of the same object. No new object or new version of the object will be created. User could also click cancel button to dismiss the dialog and no values will be saved.

2. Activity Feature Object Definition Dialog: This dialog is used for value entries for activity feature object definition. There are 8 entries in the dialog:

- Class entry: user defines class definition by selecting from a list.
- Id entry: user defines the object id by entering the id number at the input prompt. If there is an object with the id number user entered and stored in the product feature object database, the value of that object will fill in the rest of the entries, and the edit button is activated.
- Work package entry: user defines the work package value by selecting from a list. The available values in the selection list are defined by the class entry user selected. The selection could be single selection or multiple selection.
- Method entry: user defines the type value by selecting from a list. The available values in the selection list are defined by the class entry user selected.
- Subcontractor entry: user defines the subcontractor value by entering string values.

- **Process entry:** user defines the process value by selecting from a list. The list contains percentage number as: 0%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, 100%.

- ***Proceed_by* entry:** user defines the *proceed_by* value by three steps. First, user selects a class from a list. The available values in the selection list are defined by the class entry user selected. Then user enters the id number for a certain object of the class. The add button beneath the selection prompt is activated. User click the add button, the *proceed_by* object is entered into the *proceed_by* object list. User could select a *proceed_by* object from the *proceed_by* object list. The remove button above the list is activated. User click the remove button, then the selected object is removed from the *proceed_by* object list. If there is no object in the *proceed_by* object list, the remove button will be automatically turned inactive.

- ***Succeed_by* entry:** user defines the *succeed_by* value by three steps. First, user selects a class from a list. The available values in the selection list are defined by the class entry user selected. Then user enters the id number for a certain object of the class. The add button beneath the selection prompt is activated. User click the add button, the *succeed_by* object is entered into the *succeed_by* object list. User could select a *succeed_by* object from the *succeed_by* object list. The remove button above the list is activated. User click the remove button, then the selected object is removed from the *succeed_by* object list. If there is no object in the *succeed_by* object list, the remove button will be automatically turned inactive.

After necessary entries are entered, user could click new record button. The value for the object user defined will be entered into the activity feature object database as a

new object or if there are already one or more versions of the object, the values will be saved to database as a new version of the object. The relationship is built at the time object is saved into database. If there is one or more versions of object in the activity feature object database, user enter the id number for the object, the values for the latest version of the object will fill in the entries. User could edit the values. And then click the edit record button. The edited value will be saved to the database as the same version of the same object. No new object or new version of the object will be created. User could also click cancel button to dismiss the dialog and no values will be saved.

3. Resource Feature Object Definition Dialog: This dialog is used for value entries for resource feature object definition. There are 8 entries in the dialog:

- Class entry: user defines class value by selecting from a list. The list contains material, equipment and labor.
- Resource entry: user defines resource value by selecting from a list. The available values in the selection list are defined by the class entry user selected. The available list defined by material is draw from the value of product feature object database material field values. The available list defined by equipment is drawn from the joined value of method field in activity feature object database, and, equipment field in method feature object database.
- Availability entry: user defines the availability value by selecting from a list. The list includes “purchased”, “in transportation”, “in site” and “out of order”.
- Material entry: user defines the type value by selecting from a list. The available values in the selection list are defined by the class entry user selected.

- Unit of measure entry: User defines the unit of measure value by selecting from a list. The available values in the selection list are defined by the class entry user selected.

- Unit cost entry: User defines unit cost value by entering money value.
- Provider entry: User defines provider value by entering character value.
- Quantity entry: User defines quantity value by entering decimal value according to the unit of measure.

- Picture Item entry: user clicks the browsing button next to it and select a drawing item file, the filename and path of that file then is shown at the text line. This value is optional.

After necessary entries are entered, user could click new record button. The value for the object user defined will be entered into the resource feature object database as a new object or if there are already one or more versions of the object, the values will be saved to database as a new version of the object. If there is one or more versions of object in the resource feature object database, user select the resource object from the resource selection, the values for the latest version of the object will fills in the entries. User could edit the values. And then click the edit record button. The edited value will be saved to the database as the same version of the same object. No new object or new version of the object will be created. User could also click cancel button to dismiss the dialog and no values will be saved.

4. Method Feature Object Definition Dialog: This dialog is used for value entries for activity feature object definition. There are 5 entries in the dialog:

- Activity entry: user selects activity that uses the on-going defined method feature object for the field value of method by selecting from a list.

- Method entry: user selects method object value for defining from a list. The available values for the list are drawn from method field value in the activity feature object database table.

- Productivity entry: user defines the productivity value by entering decimal values.

- Equipment entry: user defines the equipment value by three steps. First, user selects an equipment type from a list. The available values in the selection list are defined by the method entry user selected. Then user enters integer value for equipment quantity. The add button beneath the selection prompt is activated. User click the add button, the equipment and quantity of that equipment needed are entered into the equipment list. User could select an equipment value from the equipment list. The remove button above the list is activated. User clicks the remove button, and then the selected value is removed from the equipment list. If there is no value in the equipment list, the remove button will be automatically turned inactive.

- Labor entry: user defines the labor value by three steps. First, user selects a labor type from a list. The available values in the selection list are defined by the method entry user selected. Then user enters integer value for labor quantity. The add button beneath the selection prompt is activated. User click the add button, the equipment and quantity of that equipment needed are entered into the equipment list. User could select an equipment value from the equipment list. The remove button above the list is activated. User clicks the remove button, then the selected value is removed from the

labor list. If there is no value in the labor list, the remove button will be automatically turned inactive.

After necessary entries are entered, user could click new record button. The value for the object user defined will be entered into the method feature object database as a new object or if there are already one or more versions of the object, the values will be saved to database as a new version of the object. If there is one or more versions of object in the activity feature object database, user select the method object from the method selection, the values for the latest version of the object will fills in the entries. User could edit the values. And then click the edit record button. The edited value will be saved to the database as the same version of the same object. No new object or new version of the object will be created. User could also click cancel button to dismiss the dialog and no values will be saved.

5. Participant Feature Object Definition Dialog: This dialog is used for value entries for participant feature object definition. There are 10 entries in the dialog:

- Class entry: user defines class value by selecting from a list. The list contains subcontractor, material provider, equipment provider and labor provider.

- Participant entry: user defines participant value by selecting from a list. The available values in the selection list are defined by the class entry user selected. The available list defined by subcontractor is draw from values of subcontractor field in activity feature object database table. The available list defined by material provider, equipment provider and labor provider is drawn from values of provider field in resource feature object database table.

- Address entry: User defines the address value by entering character value.

- Phone entry: User defines this entry by entering character value.
- Fax entry: User defines this entry by entering character value.
- Email entry: User defines this entry by entering character value.
- Contract type entry: user defines the contract type by selecting from a list.
- Contract scope entry: user defines the contract scope by selecting from a list.
- Contract no. entry: User defines this entry by entering character value.
- Contract amount entry: User defines this entry by entering money value.

After necessary entries are entered, user could click new record button. The value for the object user defined will be entered into the participant feature object database as a new object or if there is already one or more versions of the object, the values will be saved to database as a new version of the object. If there is one or more versions of object in the participant feature object database, user enter the participant object, the values for the latest version of the object will fills in the entries. User could edit the values. And then click the edit record button. The edited value will be saved to the database as the same version of the same object. No new object or new version of the object will be created. User could also click cancel button to dismiss the dialog and no values will be saved.

Cost View Window

- A. Purpose: Define the cost view objects and perform preliminary cost analysis.
- B. Composition: This windows combine map display, table display and chart display.

C. Display view: The view for each window includes a view displaying the spatial representation of objects and tables for reporting the analysis results.

D. Menu: Four menus served in cost view window:

1: File menu: same as in CAD Object Definition Window.

2: Cost analysis menu: including five menu items:

- Build Material Cost Items: is used to build material cost item themes with the feature object database.

- Build Equipment Cost Items: is used to build equipment cost item themes.

- Build Labor Cost Items: is used to build labor cost item themes.

- Calculate Cost: is used to calculate cost for each cost item. The result is added to the theme database table.

- Work Estimate: is used to summarize estimate for each work category. The subtotal estimating about material, equipment and labor used in each work is calculated and added to the result table. The total estimating is also added to the result table.

- Element Estimate: is used to summarize estimate for each product element category. The subtotal estimating about material, equipment and labor used in each element is calculated and added to the result table. The total estimating is also added to the result table.

- Contract Estimate: is used to calculate contract cost for each material, equipment or labor provider.

3. Graphics menu: same as in CAD Object Definition Window.

4. Window menu: same as in CAD Object Definition Window.

5. Help menu: same as in Project Window.

E. Tools: There are two sets of tools in cost view window: object identification tools and view navigation tools.

1. Object identification tool set: Three tools are used to identify the cost view object:

- Identification tool: is used to identify user-selected object. User first makes the interested theme the active theme and clicks on the interested object. The value of the selected object is shown on the identify window.

- Picture link tool: is used to display the picture item for user-selected object. User first makes the interested cost item theme the active theme and clicks on the interested object. The picture item for that object will be displayed in an image window. I

- Label tool: is used to label selected object. User clicks on the interested object. The object then is labeled with the object id.

- View Navigating Tool Set: This tool set is used to navigate the view. It includes three tools: Zoom in tool: is used to zoom in on the position user clicks or the box user defines on a view.

- Zoom out tool: is used to zoom out from the position user click or the area user defines on a view.

- Pan tool: is used to pan the view by user dragging it in any direction with the pan tool.

Process View Window

A. Purpose: Define the process view objects and perform project planning and as-built scheduling analysis.

B. Composition: This unit combines 3 windows, cost view window, process view window and site view window. Each window has its own set of display views, menus and tools.

C. Display view: The view for each window includes a view displaying the spatial representation of objects and tables for reporting the analysis results.

D. Menu: Four menus served in cost view window:

1: File menu: same as in CAD Object Definition Window.

2: Process analysis menu: including five menu items:

- Define Work Items: is used to build work item themes with the feature object database. The resulted cost item themes are classified according to activity feature object class. Each class of activity is defined as one process view theme. The spatial representation of these themes is built on the extent of the working item. The attributes for themes come from the database attributes of product feature object, activity feature object and method feature object.

- Show Tables: is used to open database tables associated with work items.
- Calculate Duration: is used to calculate work duration for each work item.
- Generate Schedule: is used to generate the project schedule according to work duration and the precedence relationship among works. The result is a schedule table and work flow chart.

- Show Bar Chart: is used to generate and display construction activity workflow bar chart.

- **Generate As-Built Schedule:** is used to generate the as-built timing information for each work. The as-built timing is collected through all versions of activity feature object.

3. Graphics menu: same as in CAD Object Definition Window.

4. Window menu: same as in CAD Object Definition Window.

5. Help menu: same as in Project Window.

E. Tools: Same as in Cost View Window.

Site View Window

A. Purpose: Define the site view objects and perform project progress analysis.

B. Composition: This unit combines 3 windows, cost view window, process view window and site view window. Each window has its own set of display views, menus and tools.

C. Display view: The view for each window includes a view displaying the spatial representation of objects and tables for reporting the analysis results.

D. Menu: Four menus served in cost view window:

1: File menu: same as in CAD Object Definition Window.

2: Function menu: including five menu items:

- **Build Site Items:** is used to build site item themes with the feature object database. The resulted cost item themes are classified according to product feature object class. Each class of product is defined as one site view theme. The spatial representation of these themes is built on the spatial shape definition for each individual theme. The

attributes for the themes come from the database attributes of product feature object and activity feature object.

- **Show tables:** is used to open database tables associated with site item themes.

- **Check Process:** is used to summarize current working process for each site theme. Each site theme will be categorized according to work process. The result will be shown as single legend for each process category: not_start, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, 100%, finished.

- **Check History Process:** is used to summarize working process for each site theme at the specific date user entered. First user enters a date, and then the site theme object snapshot at the date will be classified according to work process.

- **Check Specification:** is used to check product specification. First user select site item, and then the special specification will be shown in a text window.

- **Analyze Safety:** is used to analyze safety on site. The hazardous zone and accident-prone activity will be displayed on map. Safety report will be produced and shown in test window.

- **Analyze Change Order:** is used to analyze the manipulate change order. First user edits the site item that is supposed to be changed, and then the contingent change on related items will be displayed. After user confirms the change order, the change will be propagated to related items.

- **Clear View:** is used to clear the view graphics.

3. Graphics menu: same as in CAD Object Definition Window.

4. Window menu: same as in CAD Object Definition Window.

5. Help menu: same as in Project Window.

E. Tools: same as in Cost View Window.

LIST OF REFERENCES

Alexander, J. F., 1996. Gator Communicator: Design of a Hand Held Digital Data Mapper. Computing in Civil Engineering, American Society of Civil Engineers(ASCE).

Aouad et al. 1994. Integration of Construction Information. Final Report, University of Salford, Manchester, England.

Aouad G. 1997. Integration: From a modeling dream into an implementation reality. Proceedings of CIB W55, International support for building economics, Lake District, UK, pp 7-29.

Amor R. W., M. Clift, R. Scherer, P. Katranuschkov, Z. Turk and M. Hannus, 1997. A Framework for Concurrent Engineering – ToCEE, European Conference on Product Data Technology, PDT Days 1997, CICA, Sophia Antipolis, France, April, pp. 15-22.

Andresen J., A. Baldwin, B. Martin, C. Carter, A. Hamilton, E. Stokes, T. Thorpe, 2000. A Framework for Measuring IT Innovation Benefits. ITCon Vol. 5 (2000), pp. 57-72. <http://itcon.org>.

Arkinci B., M. Fischer, 1998. Time-Space Conflict Analysis Based on 4D Production Models. Proceedings of the ASCE Congress on Computing in civil Engineering, Boston, pp. 342-353.

Arkinci B., M. Fischer, 2000. An Automated Approach for Accounting for Spaces Required by Construction Activities. Proceeding of Construction Congress IV, February 2000, Orlando, Florida, pp. 1-10.

Armstrong, M.P. & P.J. Densham, 1994. Toward the Development of a Conceptual Framework for GIS-based Collaborative Spatial Decision-Making. Proceedings of the 2nd ACM Workshop on Advances in GIS, Baltimore, Maryland, December 1994.

Barrett P., 2000. Construction Management Pull for 4D CAD. Proceeding of Construction Congress IV, February 2000, Orlando, Florida, pp. 977-983.

Bjork, B.C., K. Lownertz and A. Kiviniemi, 1997. ISO DIS 13567 – The Proposed International Standard for Structuring Layers in Computer Aided Building Design. Electronic Journal of Information Technology in Construction. Electronic Journal of Information Technology in Construction, March 1997.

Bjork, B.C. 1999. Information Technology in Construction: Domain Definition and Research Issues. International Journal of Computer Integrated design and Construction, SETO, London, UK. May 1999, Vol. 1 No. 1, pp. 3-16.

Brandon, P., M. Betts and Hans Wamelink, 1998. Information Technology Support to Construction Design and Production. Computers in Industry 35, Elsevier Science B.V. Amsterdam, The Netherlands. .

Business Geography Resource Site, 1998. Business and Economic Geography Utilizing for Business Planning. <http://weber.u.washington.edu/~mcmullin>

Cattell, R.G. G., 1996. The Object Database Standard: ODMG-93, Morgan Kaufmann, San Francisco, California.

Chin, S., A. L. Stumpf and L. Y. Liu, 1996. Object-Oriented Construction Information Framework for Construction Management, Journal of Computing in Civil Engineering, ASCE.

Chrisman, N., 1997. Exploring Geographic Information Systems. New York: John Wiley and Sons Inc.

CIFE, 1999. Facilities Management: Move Management Charrette. <http://www.stanford.edu/group/CIFE/CIFE-CharretteWhitePaper.html>, 1999.

Coble R. J., D. Theisen and R. L. Blatter, 2000. Application of 4D CAD in Construction Workplace. Proceeding of Construction Congress IV, February 2000, Orlando, Florida, pp. 984-989.

Cole, M. et al, 1998. IFC Model Requirement Analysis, ES-1: Cost Estimating, IAI Project Document for IFC R2.0.

Collier E., and M. Fisher, 1995. Four Dimensional Modeling in Design and Construction. Technical Report, Nr. 101, CIFE, Stanford.

COMBINE, 1998. COMBINE 2 – Computer Models for the Building Industry in Europe. <http://erg.ucd.ie/combine.html>

Digre, T., 1998. Business Object Component Architecture, IEEE Software.

Eastman C., 1999. Information Exchange Architectures for Building Models. Proc. 8th International Conference on Durability of Building Materials and Components, Vancouver, Canada, May~June 1999.

ESRI, 2000. About GIS. <http://www.esri.com/library/gis/index.html>.

Fischer M., 1989. Design Construction Integration through Constructibility Design Rules for the Preliminary Design of Reinforced Concrete Structures. Proceedings of CSCE/CPCA Structural Concrete Conference, Montreal, Canada, 1989.

Fisher M., Hannus M., Yamazaki Y. & Latinen J., 1993. Goals, Dimensions, and Approaches for Computer Integrated Construction. In K. Mathur, M.P. Betts & K.W. Tham (eds.) Management of Information Technology for Construction, Singapore: World Scientific & Global Publications Services.

Fisher M., T. Froese & D. Phan, 1994. How Do Integration and Data Models Add Value to a Project. Computing in Civil Engineering 1994, ASCE.

Fisher, M., L. M. Waugh & A. Axworthy, 1998. IT Support of Single Project, Multiple Project and Industry-wide Integration. Computers in Industry 35, Elsevier Science B.V. Amsterdam.

Fisher M., 2000. Benefits of 4D Models for Facility Owners and AEC Service Providers. ✓
Proceeding of Construction Congress IV, February 2000, Orlando Florida, pp. 990-995.

Froese, T., 1992. Integrated Computer-Aided Project Management Through Standard Object-Oriented Models, Technical Report 68 A, Stanford University, Center for Integrated Facility Engineering (CIFE), July, 1992.

Froese, T., J. Rankin and K. Yu, 1997. Project Management Application Models and Computer-Assisted Construction Planning in Total Project Systems. International Journal of Construction Information Technology, Special Issues on Integrated environments, Vol. 5, No. 1 Summer 1997, pp. 39-62.

Froese, T., J. Rankin 1998. Representation of Construction Methods in Total Project Systems. 5th Congress on Computing in Civil Engineering, American Society of Civil Engineers (ASCE), Boston, Oct. 1998.

Froese, T., K. Yu, 1999. Industry Foundation Class Modeling for Estimating and Scheduling, Proceeding of 8th International Conference on Durability of Building Materials and Components, Vancouver, May ~ June 1999. Vol. 4, pp. 2825-2835.

Froese, T. 1999. Interwoven Threads: Trends in the Use of Information Technologies for the Construction Industry. White Paper prepared for Berkeley-Stanford Workshop on Defining a Research Agenda for AEC Process/Product Development in 2000 and Beyond, Stanford, CA, Aug. 1999.

Froese, T., M. Fischer, F. Grobler, J. Ritzenthaler, K. Q. Yu, S. Sutherland, S. Staub, B. Akinci, R. Akbas, B. Koo, A. Barron, and J. Kunz, 1999 Industry Foundation Classes for Project Management – A Trial Implementation. Electronic Journal of Information Technology in Construction, Vol. 4, 1999, pp. 17-36.

Gielingh, W. F., 1987. General Reference Model for AEC Product Definition Data (GARM) (version 3). TNO Building and Construction Research, Delft, TNO rapport BI-87-87.

Grobler, F. et al., 1998. IFC Model Requirement Analysis, PM-1: Scheduling, IAI Project Document for IFC R3.0.

Hackney, J.W. , 1991. Cost Coding. Control & management of capital project, 2nd Edition. MacGraw-Hill, New York, N.Y, 31-41.

Halfawy, M. R., F. C. Hadiprionon, J. W. Duane and R. E. Larew, 1996. Visualization of Spatial and Geometric Databases for Construction Projects. Computing in Civil Engineering, ASCE, 1996.

Hannus, M., 1992. Information Models for Performance Driven Computer Integrated Construction. In D.J. Vanier & J.R. Thomas (eds.) Joint CIB Workshops on Computer and Information in Construction, Montreal, Canada, CIB 1992.

Hannus, M. and K. Pietilainen, 1995. Implementation Concerns of Process Modeling Tools, CIB workshop on "Information Technology in Construction"(W78/TG10), California, USA, August 1995.

Hannus, M., J. Laitinen and K. Seren, 1996. Integrated Design and Planning in Building Projects – System Architecture, Implementation and Experience. In B. Kumar and A. Retik (eds.) Information Representation and Delivery in Civil and Structural Engineering Design. UK: Civil-Comp Press.

Helenius, J., R.P.Hamalainen, E. Kettunen, H. Luri, P. Miettinen, J. Mantysaari, M. Poyhonen, A. Salo and H. Vrolijk, 1998. Decision Analysis and Decision Support Systems. <http://www.hut.fi/HUT/Systems.Analysis/Research/act94re-2.html>

Hinze, J., Russell, D.B., 1995. Analysis of Fatalities Recorded by OSHA. Journal of Construction Engineering and Management, June 1995.

Hsu, C., 1996. Enterprise Integration and Modeling: The Metadatabase Approach. Kluwer Academic Publishers, Massachusetts.

I3CON Project, 1999. Intelligent Integration of Information in Construction (I3CON). <http://www.salford.ac.uk/iti/projects/commit/papers/iic/iic.html>

Industry Alliance for Interoperability (IAI), 1995. Volume III: IFC Model Exchange, International Alliance for Interoperability.

Industry Alliance for Interoperability (IAI), 1996. End User Guide to Industry Foundation Classes, Enabling Interoperability in the AEC/FM Industry, International Alliance for Interoperability.

Industry Alliance for Interoperability (IAI), 1997. IFC Object Model for AEC Projects, Specification Volume 2. International Alliance for Interoperability (IAI) 1997.

Industry Alliance for Interoperability (IAI) 1996. "End User Guide to Industry Foundation Classes, Enabling Interoperability in the AEC/FM Industry", International Alliance for Interoperability (IAI) 1996.

Industry Alliance for Interoperability (IAI), 1997. Guidelines for the Development of Industry Foundation Classes" draft, International Alliance for Interoperability (IAI), May 1997.

Industry Alliance for Interoperability (IAI) 2000, International Alliance for Interoperability (Home Page), <http://www.interoperability.com>

IntelliCorp, 1999. Knowledge-based Tools. http://www.intellicorp.com/prod_kbt.html.

International Standard Organization (ISO), 1994a. Industry Automation Systems – Product Information Representation and Exchange – Part 1, Overview and Fundamental Principles. ISO TC 184/SC4 Project Management Advisory Group.

International Standard Organization (ISO), 1994b. Building Construction Core Model. Project Proposal, ISO TC 184/SC4/WG3 document N341, October 1994.

International Standard Organization (ISO), 1994c. Classification of Information in the Construction Industry. ISO Technical Report 14177: (E). ISO Standard 10303.

International Standard Organization (ISO), 1996. ISO 10303 – AP225: Product Data Representation and Exchange – Building Elements Using Explicit Shape Representation. ISO TC 184/SC4/WG3 N510 (T12) July 1996.

Imhoff, C. & G. Battas, 1996. Building the Operational Data Store. John Wiley & Sons Inc., New York.

Industrial RTD Project (ESPRIT), 1998. ToCEE – Towards a Concurrent Engineering Environment in the Building and Engineering Structures Industry, European Union, <http://www.cib.bau.tu-dresden.de/tocee>.

Karhu, V., 1995. Design Process Models Using IDEF0. STAR research program, VTT Building Technology. <http://www.vtt.fi/cic/projects/star/star1/process/process.html>

Koo B., M. Fischer, 1998. Feasibility study of 4D CAD in Commercial Construction. Technical Report, Nr. 118, CIFE, Stanford.

Kosters, G., B.U. Pagel, and H. W. Six, 1995. Object-Oriented Requirements Engineering for GIS Applications. In Proceedings of the ACM International Conference on Geographic Information Systems.

Laitinen, J., 1995. Model Based Cost Engineering as a Basis for the Management of Construction Process. In M.A. Fisher, K.H. Law & B. Luiten (eds.) Modeling of Building Through Their Life-Cycle, CIB W78 & TG10 Workshop.

Laitinen, J. 1999. Model Based Construction Process Management. Durability of Building Materials and components 8. National Research Council of Canada: Ottawa. Vol.4, pp. 2844-2836.

Luiten, G. T., F. P. Tolman and M. A. Fisher, 1998. Project Modeling in AEC to Integrate Design and Construction, Computers in Industry 35 (1998), Elsevier Science B.V.

Lottaz C., R. Stouffs, I. Smith, 2000. Increasing Understanding during Collaboration through Advanced Representations. ITCon Vol. 5 (2000) pp. 1-24.

McKinney, K., J. Kim, M. Fischer and C. Howard, 1996. Interactive 4D-CAD, Computing in Civil Engineering, ASCE.

McKinney, K. and M. Fischer, 1997. 4D Analysis of Temporary Support, Journal of Computing in Civil Engineering, ASCE 1997.

OSCON 1997, Open Systems for CONstruction. <http://www.salford.ac.uk/iti/oscon.html>

Rackley S., L. Waugh, T. Froese and J. Rankin, 1998. Classification Hierarchies: A Proposed Graphical Interface. Proceedings of the Canadian Society for Civil Engineering Annual conference, Halifax, Canada.

Rezgui, Y. and P. Debras, 1996. An Integrated Approach for a Model Based Document Production and Management, Electronic Journal of Information Technology in Construction. <http://fitcon.org/1996/1/yacine.htm>

Rankin J. H., T. M. Froese, L.M. Waugh, 1999. Application of Case-based Reasoning to Computer-assisted Construction Planning. 8th International Conference on Durability of Building Materials and Components, May ~ June 1999, Vancouver, Canada.

Rezgui, Y., G. Cooper & P. Brandon, 1998. Information Management in a Collaborative Multi-actor Environment: The COMMIT Approach, Journal of Computing in Civil Engineering, Vol. 12 No. 3, ASCE.

Riley D., 2000. The Role of 4D Modeling in Trade Sequencing and Production Planning. Proceeding of Construction Congress IV, February 2000, Orlando, Florida, pp. 1029-1034. ✓

Rivard H., 2000. A Survey on the Impact of Information Technology on the Canadian Architecture, engineering and Construction Industry. Itcon Vol. 5 (2000) pp. 37-56

Russell, A. and T. Froese, 1997. Challenges and A Vision for Computer-Integrated Management Systems for Medium-Sized Contractors. Canadian Journal of Civil Engineering, Vol. 24, No.2, April 1997, pp. 180-190.

Russell, A. and S. AbouRizk, 1996. Automated Generation of Productivity Functions, Computing in Civil Engineering, ASCE.

Salford COMMIT, 1996. COMMIT (Construction Modeling and Methodologies for Intelligent information inTegration). <http://www.salford.ac.uk/iti/projects/commit>.

Seren, K., 1997. Internet-based Computer Integration in Building Construction – A Process View. VTT Building Technology, Espoo, Finland. <http://www.vtt.fi/cic/rtekjs/rtekjs.html>

Shekhar S., M. Coyle, B. Goyal, D.R. Liu, and S. Sarkar, 1997. Data Models in Geographic Information Systems. Communications of the ACM, April 1997/Vol. 40. No. 4.

Staub S., M. Fischer and M. Spradlin, 1999. Into the Fourth Dimension. Civil Engineering, ASCE 69(5), pp. 44-47.

Stefanakakis, E. & T. Sellis, 1998. Enhancing Operations with Spatial Access Methods in a Database Management System for GIS. Cartography and Geographic Information Systems, Vol. 25, No. 1, 1998.

Stumpf A. L., et. al, 1996. Object-Oriented Model for Integrating Construction Product and Process Information. Journal of Computing in Civil Engineering, ASCE, 10(3).

Tryfona, N. and T. Hadzilacos, 1995. Geographic Application Development: Models and Tools for the Conceptual Level. In Proceedings of the ACM International Conference on Geographic Information System.

Wix, J., 1998. Purpose of a Core Model. <http://helios.bre.co.uk/ccit/info/bccm/purpose>

Worboy, M.F., 1994. Object-oriented Approaches to Geo-referenced Information. Int. J. of Geographic Information System 8, 4, 1994.

Wright, J.R., 2000. GIS and Civil Engineering. <http://www.wmich.edu/jcce/e731.htm>

Yamazaki Y. and J. Maeda, 1998. The SMART System: an Integrated Application of Automation and Information Technology in Production Process, Computers in Industry 35, Elsevier Science B.V. Amsterdam.

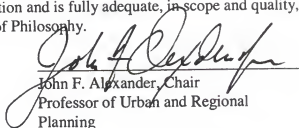
Yu, K. and F. Grobler, 1998. International Alliance For Interoperability: IFCs. Computing in Civil Engineering: Proc. Of the International Computing Congress, ASCE, Boston, MA. Pp. 395-406.

Yu K., T. Froese and F. Grobler, 2000. A Development Framework for Data Models for Computer-Integrated Facilities Management, Automation in Construction, special issue on facilities management, 2000.

BIOGRAPHICAL SKETCH

Wei Sun was born in Harbin, Heilongjiang, China. She completed her Bachelor of Science degree in Urban and Regional Planning at Harbin University of Architecture and Engineering in July 1992. She went to graduate school at Harbin University of Architecture and Engineering, Heilongjiang, China, in August 1993 and finished her Master of Science degree in Urban and Regional Planning in July 1994. In August 1997, she went to University of Florida to pursue her Ph.D. degree in College of Design, Construction and Planning, specializing in GIS field.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



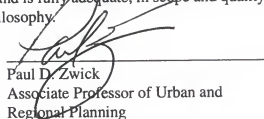
John F. Alexander, Chair
Professor of Urban and Regional
Planning

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



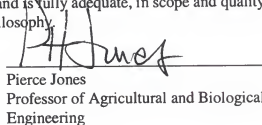
Jimmie Hinze
Professor of Building Construction

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



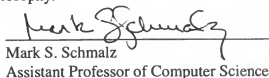
Paul D. Zwick
Associate Professor of Urban and
Regional Planning

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



Pierce Jones
Professor of Agricultural and Biological
Engineering

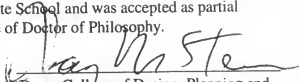
I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.



Mark S. Schmalz
Assistant Professor of Computer Science

This dissertation was submitted to the Graduate Faculty of the College of Design, Planning and Construction and to the Graduate School and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

December 2000



Dean, College of Design, Planning and
Construction

Dean, Graduate School